

“BABEŞ-BOLYAI” UNIVERSITY CLUJ-NAPOCA
FACULTY OF ECONOMICAL STUDIES
BUSINESS INFORMATION SYSTEMS DEPARTMENT

PHD THESIS

“EMPLOYING ARTIFICIAL INTELLIGENCE IN FINANCE AND BANKING”

Supervisor:
Prof. GHIŞOIU NICOLAE, PhD

PhD Candidate:
BOLOGA CRISTIAN SORIN

Cluj-Napoca, 2005

VISIT...

LANZAROTE
Caliente.COM

CUPRINS

<u>CAPITOLUL 1. INTRODUCERE</u>	6
<u>CAPITOLUL 2. STADIUL CUNOAȘTERII</u>	10
2.1. SCURT ISTORIC AL INTELIGENȚEI ARTIFICIALE.....	10
2.2.OBIECTIVELE CERCETĂRII ÎN DOMENIUL INTELIGENȚEI ARTIFICIALE ...	14
2.3. INTRODUCERE ÎN DOMENIUL PIEȚEI DE CAPITAL	18
2.3.1. <i>Inteligența artificială în economie</i>	18
2.3.2. <i>Aspecte privind domeniul financiar –bancar</i>	19
2.3.3. <i>Locul și rolul piețelor de capital în cadrul piețelor financiare</i>	20
2.3.4. <i>Politici investiționale</i>	23
2.3.5. <i>Caracteristicile investițiilor în valori mobiliare</i>	24
2.3.6. <i>Concluzii</i>	25
<u>CAPITOLUL 3. ANALIZA PIEȚEI DE CAPITAL</u>	27
3.1. ANALIZA FUNDAMENTALĂ	28
3.1.1. <i>Generalități</i>	28
3.1.2. <i>Indicatori folosiți în analiza cursului acțiunilor</i>	29
3.1.3. <i>Estimarea valorii intrinseci pe baza actualizării veniturilor viitoare</i>	34
3.1.4. <i>Analiza statistică a rentabilității și riscului valorilor mobiliare</i>	35
3.1.5. <i>Concluzii privind analiza fundamentală</i>	40
3.2. ANALIZA TEHNICĂ	41
3.2.1. <i>Generalități</i>	41
3.2.2 <i>Tendențe și tunele; cap și umeri; Formațiile M și W</i>	41
3.2.3. <i>Teoria Dow</i>	44
3.2.4. <i>Analiza cu ajutorul mediilor mobile</i>	45
3.2.5. <i>Indicatorii urmăriți de analiștii tehnici</i>	47
3.2.6. <i>Concluzii privind analiza tehnică</i>	48
3.3. TEORIA PIEȚELOR EFICIENTE (EFFICIENT MARKET HYPOTHESIS -EMH)..	49
3.3.1. <i>Generalități</i>	49
3.3.2. <i>Teoria “pieței la întâmplare” (“random walk”)</i>	50
3.3.3. <i>Concluzii privind teoria piețelor eficiente</i>	52
3.4. ANALIZA CURSULUI ACȚIUNILOR ȘI A OBLIGAȚIUNILOR	53
3.4.1. <i>Variabilitatea rentabilității acțiunilor individuale</i>	53
3.4.2. <i>Volatilitatea unei acțiuni</i>	53
3.4.3. <i>Riscul unei acțiuni</i>	55
3.4.4. <i>Valoarea actualizată a unei obligațiuni</i>	55
3.4.5. <i>Volatilitatea unei obligațiuni</i>	56
3.4.6. <i>Durata unei obligațiuni</i>	58
3.4.7. <i>Riscul unei obligațiuni</i>	58
3.4.8. <i>Concluzii</i>	59

<u>CAPITOLUL 4. PROBLEMATICA INTELIGENȚEI ARTIFICIALE</u>	61
4.1. SISTEME INTELIGENTE.....	61
4.1.1. <i>Structura sistemelor inteligente</i>	61
4.1.2. <i>Caracteristici ale sistemelor inteligente</i>	61
4.1.3. <i>Etape în rezolvarea unei probleme folosind sisteme inteligente</i>	62
4.1.4. <i>Rețelele neuronale</i>	63
4.1.5. <i>Algoritmii genetici</i>	65
4.1.6. <i>Sistemele fuzzy</i>	67
4.1.7. <i>Sistemele expert</i>	69
4.1.8. <i>Agenții inteligenți</i>	70
4.1.9. <i>Sistemele hibride</i>	71
4.2. SISTEME DE ASISTARE A DECIZIILOR.....	73
4.2.1. <i>Noțiuni despre procesul decizional</i>	73
4.2.1.1. <i>Noțiuni introductive</i>	73
4.2.1.2. <i>Rolul sistemului informațional</i>	75
4.2.1.3. <i>Decizia în managementul firmei</i>	76
4.2.1.4. <i>Tipuri de decizii</i>	78
4.2.1.5. <i>Tratarea incertitudinii</i>	80
4.2.1.6. <i>Tratarea riscului</i>	81
4.2.1.7. <i>Etapele procesului decizional</i>	83
4.2.1.8. <i>Principalele categorii de instrumente informatice folosite în procesul decizional</i>	84
4.2.2. <i>Conceptul de Sistem de asistare a deciziilor</i>	86
4.2.2.1. <i>Noțiuni introductive</i>	86
4.2.2.2. <i>Caracteristicile sistemelor de asistare a deciziilor</i>	89
4.2.2.3. <i>Structura unui sistem de asistare a deciziilor</i>	91
4.2.2.4. <i>Fazele dezvoltării unui sistem de asistare a deciziilor</i>	92
4.2.3. <i>Concluzii</i>	92
4.3. SISTEME EXPERT	93
4.3.1. <i>Privire generală asupra sistemelor expert</i>	93
4.3.1.1. <i>Scurtă introducere în sisteme expert</i>	93
4.3.1.2. <i>Necesitatea utilizării sistemelor expert</i>	97
4.3.1.3. <i>Caracteristicile sistemelor expert</i>	99
4.3.1.4. <i>Structura unui sistem expert</i>	104
4.3.1.5. <i>Tipuri de sisteme expert</i>	106
4.3.1.6. <i>Funcțiile unui sistem expert</i>	108
4.3.1.7. <i>Roluri în cadrul unui sistem expert</i>	109
4.3.1.8. <i>Baza de cunoștințe</i>	111
4.3.1.9. <i>Achiziția cunoștințelor</i>	113
4.3.1.10. <i>Reprezentarea cunoașterii</i>	117
4.3.1.11. <i>Reprezentarea regulilor</i>	121
4.3.1.12. <i>Mașina de inferență</i>	122
4.3.1.13. <i>Mecanismele de inferență ale unui sistem expert</i>	124
4.3.1.14. <i>Motorul de elaborare și învățarea</i>	130

4.3.1.15. Utilizatorii unui sistem expert	131
4.3.1.16. Sisteme expert având la bază fapte și reguli incerte	132
4.3.2. <i>Elaborarea unui sistem expert</i>	137
4.3.2.1. Considerații generale despre elaborarea unui sistem expert	137
4.3.2.2. Limbajele de programare folosite pentru elaborarea sistemelor expert	139
4.3.2.3. Metodologii de elaborare a sistemelor expert	144
4.3.2.4. Etapele realizării unui sistem expert folosind prototipizarea	154
4.3.2.5. Ciclul de viață al unui sistem expert	158
4.3.2.6. Utilitatea și limitările sistemelor expert	161
4.3.2.7. Impactul utilizării sistemelor expert asupra activității unei firme	162
4.3.3. <i>Generatoare de sisteme expert</i>	165
4.3.3.1. Medii și generatoare pentru sistemele expert	165
4.3.3.2. Exemple de generatoare de sisteme expert	168
4.3.3.3. Etapele dezvoltării unui sistem expert cu ajutorul generatorului de sisteme expert	169
4.3.4. <i>Aplicații ale sistemelor expert în domeniul bursier</i>	170
4.3.4.1. Prezentarea unor sisteme expert; structură și mod de utilizare	170
4.3.4.2. Utilizarea sistemelor expert în domeniul detectării fraudelor pe piața bursieră	182
4.3.4.3. MarketMind	184
4.3.4.4. Investment Advisor System	185
4.3.4.5. Integrative Stock System.....	185
4.3.4.6. Integrative Expert System	186
4.3.4.7. Signal	187
4.3.4.8. ITA Intelligent Trading Assistants	188
4.3.4.9. ZETA Real Time Securities Trading Assistant	188
4.3.4.10. MIRI Mitsubishi Research Institute	189
4.3.4.11. Fund Manager	189
4.3.5 <i>Concluzii</i>	189
<u>CAPITOLUL 5. CONCEPEREA, PROIECTAREA ȘI IMPLEMENTAREA UNUI SISTEM DE ASISTARE A DECIZIILOR ÎN DOMENIUL ANALIZEI BURSIERE</u>	192
5.1. “STOCKASSIST” - SISTEM DE ASISTARE A DECIZIILOR ÎN DOMENIUL ANALIZEI BURSIERE -	192
5.2. SITUAȚIILE FURNIZATE	194
5.3. STRUCTURA FIȘIERELOR DIN BAZA DE DATE	195
5.4. INTERFAȚA SISTEMULUI	196
5.5. PROIECTAREA MODULELOR	199

5.6. APRECIERI PRIVIND EFICIENȚA ȘI FIABILITATEA SISTEMULUI INFORMATIC	202
5.7 INTEGRAREA SISTEMULUI DE ASISTARE A DECIZIILOR CU UN SISTEM EXPERT	205
<u>CAPITOLUL 6. CONCEPEREA, PROIECTAREA ȘI IMPLEMENTAREA UNUI PROTOTIP DE GENERATOR DE SISTEME EXPERT ȘI A UNUI SISTEM EXPERT ÎN DOMENIUL BURSIER</u>	207
6.1. PRODUSUL SOFTWARE UTILIZAT ÎN ELABORAREA GENERATORULUI DE SISTEME EXPERT	207
6.2. APLICAȚIA “EXPERT GENERATOR” –ARHITECTURĂ ȘI CARACTERISTICI GENERALE	213
6.3. BAZA DE CUNOȘȚINȚE A APLICAȚIEI	215
6.4. IMPLEMENTAREA APLICAȚIEI	216
6.5. UTILIZAREA APLICAȚIEI “EXPERT GENERATOR” PENTRU CREAREA UNUI SISTEM EXPERT ÎN DOMENIUL BURSIER	217
6.6. APRECIERI PRIVIND EFICIENȚA SISTEMELOR EXPERT	219
<u>CAPITOLUL 7. CONTRIBUTII PROPRII</u>	222
<u>CAPITOLUL 8. CONCLUZII FINALE ȘI PROPUNERI</u>	225
8.1. CONCLUZII.....	225
8.2. DIRECȚII DE VIITOR ALE CERCETĂRII	230
BIBLIOGRAFIE.....	233
ANEXE	241
REZUMAT AL TEZEI DE DOCTORAT	293

LISTA FIGURILOR

Fig. 1 Structura tezei de doctorat	8
Fig. 2 Testul Turing	12
Fig. 3 Domeniile Inteligenței Artificiale	16
Fig. 4 Modelul pieței bursiere	22
Fig. 5 Legea normală redusă de probabilitate	39
Fig. 6 Tendințe și tunele	42
Fig. 7 Cap și umeri în cele două configurații	42
Fig. 8 Formația M	43
Fig. 9 Formația W	43
Fig. 10 Teoria lui Dow	44
Fig. 11 Analiza cu ajutorul mediei mobile	46
Fig. 12 Evoluția cursului unei acțiuni conform teoriei “Random Walk”	51
Fig. 13 Dreapta de regresie în cazul acțiunilor foarte volatile și respectiv puțin volatile	54
Fig. 14 Procesele de fuzzyficare și defuzzyficare	68
Fig. 15 Legăturile dintre sistemele informaționale	76
Fig. 16 Procesul de luare a unei decizii	77
Fig. 17 Etapele procesului decizional	84
Fig. 18 Modelul conceptual al unui sistem de asistare a deciziilor	91
Fig. 19 Relația dintre sist. de inteligență artificială–sist. bazate pe cunoștințe–sist. expert	95
Fig. 20 Relația procesare date –procesare cunoștințe	101
Fig. 21 Arhitectura unui sistem expert	106
Fig. 22 Roluri în cadrul unui sistem expert	110
Fig. 23 Procesul de achiziție a cunoștințelor	114
Fig. 24 Fluxul de prelucrări în cadrul metodelor manuale de achiziție a cunoștințelor	115
Fig. 25 Fluxul de prelucrări în cadrul metodelor semiautomate de achiziție a cunoștințelor	116
Fig. 26 Fluxul de prelucrări în cadrul achiziției automate a cunoștințelor prin metoda inducției	116
Fig. 27 Raționarea înainte și raționarea înapoi	123
Fig. 28 Etapele ciclului de bază al unui motor de inferențe	126
Fig. 29 Limbaje folosite pentru elaborarea sistemelor expert	140
Fig. 30 Soluții de dezvoltare a sistemelor expert	144
Fig. 31 Scopul unei metodologii	144
Fig. 32 Conceptele cu care operează o metodologie	145
Fig. 33 Ciclul de viață al unui sistem expert folosind metodologii diagramate	147
Fig. 34 Metodologia focalizată	148
Fig. 35 Modelele metodologiei CommonKADS	151
Fig. 36 Schema de lucru generală în cadrul metodologiei ORSA	152
Fig. 37 Metodologia Rude pentru sisteme inteligente	153
Fig. 38 Modelul liniar al structurii ciclului de viață al unui sistem expert	160
Fig. 39 Realizarea evolutivă a sistemelor expert	160
Fig. 40 Factori în alegerea unui generator de sisteme expert	168
Fig. 41 Etapele dezvoltării unui sistem expert cu ajutorul generatorului de sisteme expert	170
Fig. 42 Structura bazei de date a aplicației “StockAssist”	195
Fig. 43 Structura meniului aplicației ”StockAssist”	197
Fig. 44 Integrarea sistemului de asistare a deciziilor cu un sistem expert	206
Fig. 45 Arhitectura aplicației “Expert Generator”	213
Fig. 46 Structura bazei de cunoștințe a aplicației “Expert Generator”	216

CAPITOLUL 1

INTRODUCERE

Mediul de afaceri românesc se dezvoltă sub semnul instabilității și al incertitudinii. Persoanele implicate în acest mediu iau deseori decizii pe baza intuiției și a “bunului-simț”, bazându-se pe o interpretare personală a fenomenelor și proceselor economice, iar rezultatele negative sunt percepute ca rezultat al destinului. În cea mai mare parte însă, aceste rezultate negative sunt efectul lipsei de informații, a lipsei unui sistem care să permită organizarea, sistematizarea, consultarea și analiza datelor existente.

Într-un mediu abundent în informații, organizația performantă trebuie să-și dezvolte capacitatea de a selecta, gestiona, prelucra și folosi inteligent *informația*. Numărul mare de alternative decizionale existente în prezent, costul mare al unor posibile disfuncționalități în condițiile unei concurențe acerbe precum și timpul de reacție deosebit de scurt impus de condițiile mediului actual de afaceri ne conduc spre ideea că funcționarea unei organizații, a unei firme, nu este posibilă fără existența și utilizarea eficientă a unui sistem informațional. De regulă, la nivelul unei entități economice, sistemul informațional este compus din subsisteme informaționale care corespund cel mai adesea compartimentelor mari ale unei unități economice: cercetare-dezvoltare, financiar-contabil, producție, logistică, comercial, resurse umane etc. În general acestea comunică între ele și fiecare subsistem informațional include subsistemul informatic corespunzător din sistemul informatic al firmei.

Un *sistem informațional* primește resurse (date) reprezentând intrări și le prelucurează prin transformare în produse finite (informații) ca ieșiri. Rolul său este de asigurare a legăturii între sistemul conducător și sistemul condus, între elementele materiale și cele umane ale unei organizații. Astăzi, în societatea informațională în care trăim, societate bazată pe informație și cunoaștere, succesul unei organizații depinde nemijlocit de modul în care este conceput, organizat și folosit sistemul informațional. Fundamentarea științifică a deciziilor manageriale, de la cele strategice la cele operative, se face de obicei pe baza unui volum foarte mare de informații. Sistemul decizional este prin urmare dependent de sistemul informațional, iar calitatea sistemului informațional este determinantă pentru calitatea deciziilor luate într-un mediu economic.

Sistemul informatic este partea dintr-un sistem informațional în care prelucrările se realizează cu ajutorul mijloacelor tehnologiei informațiilor (*IT Information Technology*). Acesta cuprinde culegerea, înregistrarea, stocarea, prelucrarea și transmiterea pe cale electronică a datelor și informațiilor, precum și mijloacele utilizate pentru realizarea acestora. Orice sistem informatic există numai în cadrul unui sistem informațional care îl cuprinde și îl subordonează funcțional, utilizându-l ca infrastructură a sa.

În cadrul organizațiilor economice, sistemele informatice au un puternic impact asupra activității economice, creșterea complexității mediului economic, amplificarea interacțiunilor dintre entități conducând la obligativitatea existenței unui astfel de sistem. Implementarea unui sistem informatic performant constituie o decizie strategică cu profunde implicații în eficientizarea activității economice, în creșterea calității produselor și serviciilor, în creșterea cifrei de afaceri și a cotei de piață.

Informatica financiar-bancară este o parte delimitată funcțional din informatica de gestiune, cu rol de automatizare a gestiunii patrimoniului și a operațiunilor derulate de organisme financiar-bancare.

Procesul de re tehnologizare care se desfășoară la noi în țară vizează și implementarea noilor tehnologii apărute în domeniul prelucrării informațiilor, și anume cele care privesc *inteligenta artificială*. În acest context se înscrie și lucrarea de față având ca *obiectiv principal* definirea unui cadru general privind posibilitățile de utilizare ale tehnologiilor inteligenței artificiale, în special ale sistemelor de asistare a deciziilor și ale sistemelor expert, într-unul din domeniile financiar-bancare și anume în cel al pieței de capital. Teza încearcă să studieze și impactul pe care inteligența artificială îl are asupra domeniului cercetat. Acest domeniu este unul dificil și supus erorilor, deseori la baza deciziilor stând aprecieri subiective. De aceea, el de multe ori nu poate fi abordat decât parțial folosind tehnologiile informatice convenționale. Încercăm să demonstrăm că o soluție informatică adecvată acestui domeniu poate fi totuși furnizată numai utilizând noile tehnologii ale inteligenței artificiale, respectiv sistemele de asistare a deciziei și sistemele expert.

Elaborarea acestei teze de doctorat s-a bazat pe studiul teoretic al analizei pieței financiare și al domeniului inteligenței artificiale, urmat apoi de dezvoltarea a două sisteme informatice în domeniul

bursier și anume: o aplicație de asistare a deciziei și un prototip de generator de sisteme expert, ce poate fi utilizat în mai multe domenii. În acest context considerăm că lucrarea de față se constituie într-un studiu sistematic al activității de elaborare a sistemelor informatice folosind metode ale inteligenței artificiale și se află la intersecția a mai multor domenii de cercetare: domeniul financiar-bancar, în particular cel al piețelor financiare, cel al tehnologiei de elaborare a sistemelor de asistare a deciziilor, a teoriei deciziei în condiții de risc și incertitudine, al tehnologiei sistemelor expert.

În figura 1, încercăm să surprindem structura tezei de doctorat cu reliefarea legăturilor dintre capitolele ei.

Prezentul capitol este unul de introducere în tematica cercetării și de fixare a obiectivelor tezei de doctorat.

Capitolul doi prezintă câteva aspecte din stadiul cunoașterii în cele două mari domenii care se întrepătrund în această lucrare: piața de capital și inteligența artificială. Acest capitol este doar unul de introducere în cercetare, noțiunile fiind dezvoltate în următoarele două capitole.

Capitolul trei este rezervat analizei pieței de capital, componentă a pieței financiare, accentul punându-se pe studiul indicatorilor folosiți în analiza bursieră, indicatori care constituie baza oricărei decizii de investiție. Acești indicatori vizează evaluarea financiară a valorilor mobiliare precum și a rentabilității și situației economico-financiare a societăților emitente și a economiei în ansamblu și vor fi urmăriti și în cadrul aplicației dezvoltată în capitolul cinci. Urmărirea atentă și analiza acestor indicatori pot să constituie premisa unei investiții profitabile pe piața de capital.

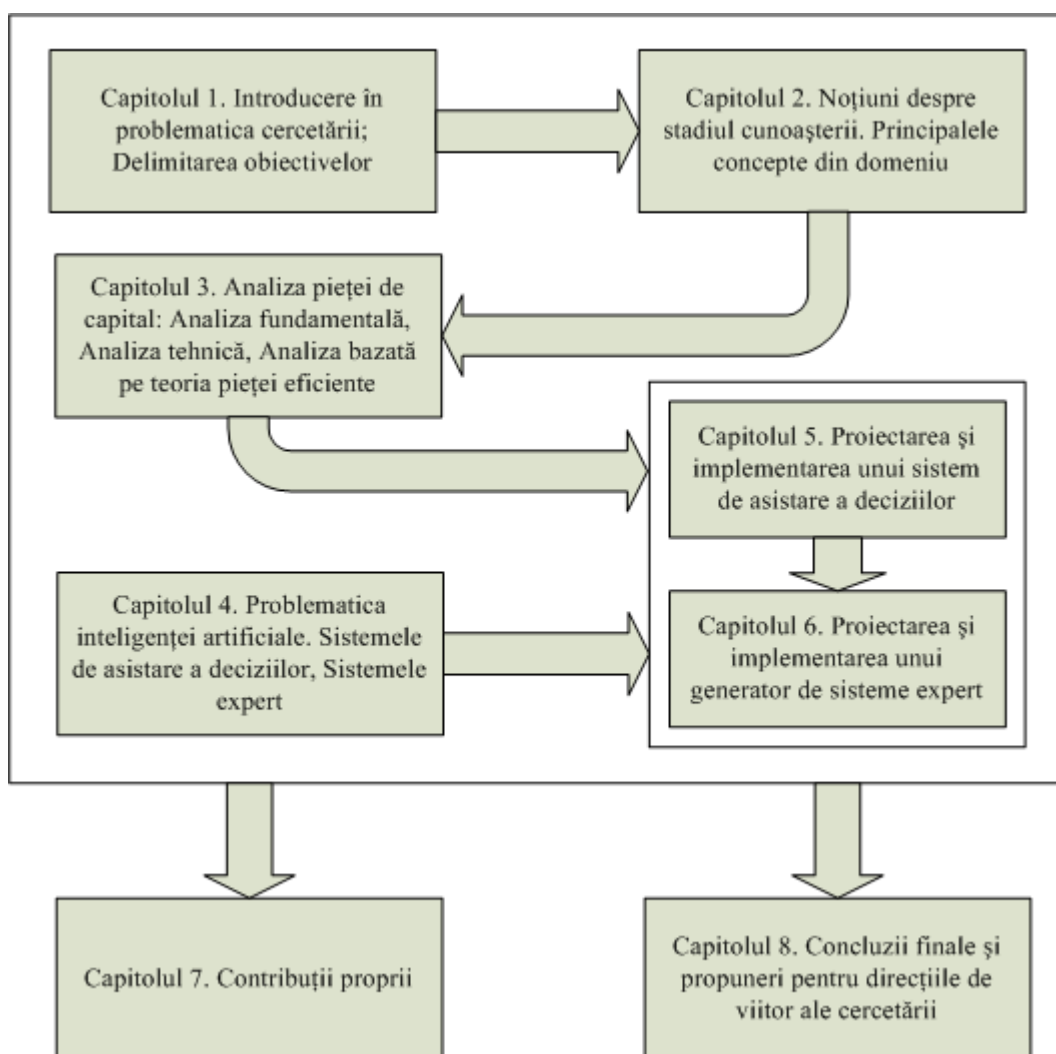


Fig. 1 Structura tezei de doctorat

Capitolul patru surprinde problematica inteligenței artificiale, prin prezentarea sistemelor de asistare a deciziilor și a sistemelor expert. Sistemele de asistare a deciziilor sunt produse program ce furnizează informații decidenților dintr-un anumit domeniu de activitate prin intermediul unor modele analitice de decizie și prin accesarea unor baze de date în scopul asigurării suportului activității decizionale în cazul problemelor complexe greu structurabile. Obiectivul principal al acestor sisteme îl constituie îmbunătățirea calității deciziilor prin realizarea unui studiu pregătitor în vederea adoptării deciziei. Dezvoltarea unui astfel de sistem a făcut obiectul capitolului cinci al prezentei lucrări. Pentru depășirea limitelor sistemelor de asistare a deciziilor, referitor la administrarea cunoștințelor și la efectuarea de raționamente cu aceste cunoștințe, am abordat tehnici specifice Inteligenței Artificiale, în particular sistemele expert. Aceste sisteme sunt destinate rezolvării unor sarcini precise, iar procedurile utilizate sunt interpretate ca o modelare a expertizei specialiștilor, respectiv a experților. Pentru modelarea și formalizarea cunoașterii, în vederea utilizării ei de către sistemele expert, este necesară cunoașterea profundă a metodelor de achiziție și reprezentare a cunoștințelor, a avantajelor și dezavantajelor fiecăreia dintre ele, în vederea selectării celor mai adecvate metode pentru obținerea unor performanțe ridicate. În cadrul acestui capitol am studiat componentele arhitecturale și principiile constructive ale sistemelor expert. Considerăm că prezintă o mare importanță utilizarea generatoarelor ca instrumente de dezvoltare a sistemelor expert. De asemenea am prezentat câteva din cele mai cunoscute sisteme expert în domeniul bursier.

Capitolul cinci prezintă o aplicație realizată de către autor, a cărei scop este asistarea deciziilor în domeniul analizei bursiere. Aceasta a fost elaborată în mediul de programare Visual FoxPro și urmărește asistarea în vederea luării deciziilor de investiții de către brokeri, traderi sau investitori ce doresc să utilizeze această aplicație. Menționăm că aplicația nu va înlocui total munca analistului, ci doar va furniza informații pe baza cărora utilizatorii să poată lua o decizie oportună, rațională și bine fundamentată teoretic.

O a doua aplicație, de data aceasta din categoria sistemelor expert, este prezentată în *capitolul șase* și reprezintă de asemenea în totalitate contribuția proprie a autorului în cadrul tezei de doctorat. Acesta se constituie într-un prototip de generator de sisteme expert elaborat în limbajul Prolog, generator ce poate fi folosit în mai multe domenii. Generatorul permite atât crearea de baze de cunoștințe pentru diverse domenii cât și identificarea unei anumite entități pe baza cunoștințelor introduse. Este exemplificată elaborarea unui sistem expert în domeniul bursier folosind acest generator. Sistemul expert exemplificat realizează identificarea unui produs financiar pe baza caracteristicilor sale.

Capitolul șapte evidențiază contribuțiile personale ale autorului din această teză de doctorat, iar *capitolul opt* formulează concluziile finale din cercetarea întreprinsă, direcțiile în care subiectele abordate pe parcursul realizării tezei vor fi completate și dezvoltate, precum și activitățile de cercetare teoretică și experimentală ce se intenționează a fi desfășurate ca urmare a rezultatelor obținute în prezenta lucrare.

Având în vedere complexitatea și vastitatea temei propuse, lucrarea de față nu își propune să epuizeze perspectivele de abordare a tematicii enunțate prin titlu. Ea reprezintă un demers de jalonare, problematizare dar și exemplificare într-un domeniu de mare importanță, care va influența în viitor, într-o manieră decisivă, realitățile economice românești și mondiale.

CAPITOLUL 2. STADIUL CUNOAȘTERII

2.1. SCURT ISTORIC AL INTELIGENȚEI ARTIFICIALE

Din punct de vedere istoric, apariția termenului de *inteligență artificială* (I.A) este legată de anul 1956 când în SUA profesorul J. McCarthy de la Massachusetts Institute of Technology (MIT Institutul Tehnologic din Massachusetts) a introdus acest termen cu prilejul întrunirii "Dartmouth Summer Research Project on Artificial Intelligence" de la Colegiul din Dartmouth (Hanover New Hampshire) a unor specialiști americani de frunte din domeniul științelor legate de teoria și practica cercetării proceselor de calcul. La aceasta întâlnire care, în Statele Unite este denumită prima conferință pe teme de inteligență artificială, au fost formulate două obiective principale în domeniul științifico-tehnic: explicarea mecanismului gândirii umane și construirea unei mașini electronice, prin intermediul căreia să se poată imita acest proces. La acest congres a fost prezentat un program de demonstrare a logicii propozițiilor intitulat "Logic Theorist" de către Newell, Shaw și Simon de la Systems Research Laboratory (RAND Corporation). Acesta putea să demonstreze teoreme matematice din primul capitol al lucrării "Principia Mathematica" a lui Whitehead și Russell. Prin aceasta s-a realizat saltul de la programele algoritmice la cele euristice, autorii definind euristica drept proces care poate rezolva problema dată, dar nu oferă garanția rezolvării. În anii 1959-1960, Newell, Shaw și Simon au realizat primele programe de demonstrare a teoremelor bazate pe logica propozițiilor ("General Problem Solver").

Multă vreme cercetarea în domeniu a avut în principal un caracter fundamental și mai puțin aplicativ. Primele sisteme nu au părăsit incintele laboratoarelor. Mulți oameni și multe instituții au renunțat la inteligența artificială, considerându-le instrumente neaplicabile în practică. Străpungerea s-a realizat după ce oamenii de știință au început să privească fiecare problemă ca temă separată, rezolvându-le individual. În anii 1960-1970 s-au dezvoltat acele sisteme cu baze de cunoștințe care erau aplicabile pe câte un domeniu distinct.

Sistemul expert DENDRAL este unul dintre primele exemple de sisteme bazate pe cunoștințe. El a fost construit la Stanford University de către B. Buchanan, J. Lederberg și E. Feigenbaum. DENDRAL era un sistem expert capabil să sintetizeze structura moleculelor organice pe baza formulelor chimice și a spectogramelor de masă. E. Feigenbaum, plecând de la ideile lui Newell, Simon și alții, a avut ideea să separe raționamentele și cunoștințele lui Lederberg de restul mecanismului informatic și astfel a rezultat primul sistem expert. Deci programul DENDRAL se consideră a fi primul sistem expert. Sistemul MYCIN, sistem expert pentru elaborarea unui diagnostic preliminar în cazul infecțiilor bacteriene ale sângelui, a cărui dezvoltare a început la Stanford University de către Edward Shortliffe și Stanley Cohen (în postura de ingineri de cunoștințe) și Stanton Axline (fizician, în calitate de expert) în jurul anilor '74-'75, este un alt exemplu de sistem care a pus în evidență rolul important al cunoștințelor specifice domeniului. În același timp, sistemul MYCIN a impus una dintre caracteristicile principale ale sistemelor bazate pe cunoștințe: capacitatea de explicare în limbaj natural a raționamentului efectuat în rezolvarea problemelor. DENDRAL a fost realizat în LISP și FORTRAN, iar MYCIN a fost elaborat în LISP în prima versiune. Acest sistem este urmat de sistemul PROSPECTOR elaborat de Duda, Hart și Nilsson, destinat prospecțiunilor geologice.

Tematica inteligenței artificiale se compunea din mai multe domenii mai mici de cercetare, dezvoltare, aplicare, care se puteau controla mai bine. La sfârșitul anilor '70, urmând succesul înregistrat mai ales de către sistemul medical MYCIN, s-au relansat cercetările în domeniul acestor sisteme. În ciuda ecurilor favorabile în cadrul cercurilor științifice, aceste sisteme mari nu au reușit să pătrundă în viața comercială. Realizarea sistemelor expert era o activitate încă mult prea complicată, nu se dispunea de unelte suficient de corespunzătoare și confirmate. Elaborarea acestora necesita efortul susținut și de durată al unui colectiv destul de mare de specialiști, cu sacrificii financiare considerabile și cu rezultate de multe ori contestate.

Abia prin anii '80 au început să apară aplicațiile comercializabile în domeniul Inteligenței Artificiale, cum ar fi generatoarele de sisteme expert; un astfel de sistem a fost KES (Knowledge Engineering System) produs de Software A&E, din care s-a realizat MICRO-PS. "MICRO" vine să sugereze faptul că pachetul este destinat microcalculatoarelor, iar "PS" este prescurtarea de la "*production system*" (sistem de producție). Unul din avantajele generatorului de sisteme expert este acela că permite un mod de construire asemănător cu limbajul natural (cel englez). Regulile apropiate de limbajul natural

permit autorului sistemului expert (*knowledge base author*) să-și îndrepte mai mult atenția asupra problemei de rezolvat și să nu se preocupe de problemele limbajului complex de programare care și el ar necesita o atenție și mai mare. Autorul nu trebuie să dispună astfel de experiență în domeniul programării. Acest fapt este un avantaj deosebit, deoarece generatoarele de genul lui MICRO-PS nu necesită ca autorul sistemului să verifice de fiecare dată programul. Mecanismul de comandă inclus în generatorul de sistem scutește autorul programului de redactarea unei părți foarte mari a codului programului. Acest mecanism numit *motor (mașină) de inferență* folosește regulile scrise de către specialistul din domeniul cunoașterii pentru a stabili ce anume trebuie să-l întrebe pe utilizator. Specialistul din domeniul cunoașterii nu trebuie să se preocupe nici de modul în care vor apare întrebările adresate utilizatorului pe ecranul calculatorului, lucrul acesta fiind realizat tot de către generatorul de sistem, de regulă prin interfețe grafice prietenoase (GUI). Gestionând și această parte a intrărilor și ieșirilor, generatorul de sistem oferă practic un mediu de lucru (shell) atât pentru crearea, dezvoltarea cât și pentru consultarea și exploatarea sistemului expert.

După succesul sistemelor cu baze de cunoștințe mari, au fost dezvoltate sisteme expert mai mici. Există păreri care consideră că prin crearea acestor sisteme mici se realizează o îndepărtare de inteligența artificială și se obțin niște sisteme care nu corespund deloc acestei noțiuni. Din această cauză ar fi nevoie probabil de o definiție a noțiunii de inteligență.

Considerăm că *inteligența* este capacitatea de a rezolva probleme, creativitatea, comunicarea cu alții sau (printr-o exprimare mai tradițională) propria convingere că acțiunea conștientă ne face inteligenți și ne deosebește de lumea animalelor? Când putem afirma despre o mașină că acționează inteligent? Dacă dorim, putem utiliza *Testul Turing* propus în anul 1950 de către matematicianul englez Alan Turing, un pionier al domeniului inteligenței artificiale și al tehnicii de calcul.

Testul Turing (ilustrat în figura nr. 2) pornește de la un joc, creat tot de Turing - ``jocul imitației’’ – cu trei jucători: o mașină (A), un om (B) și un al doilea om (C). A și B nu se găsesc în aceeași cameră cu C. Acesta nu știe care dintre ceilalți doi jucători este mașina și nu poate să îi vadă sau să le vorbească direct. Comunicarea se poate face în scris sau printr-un terminal. Scopul lui C este să deosebească mașina de om, pe baza răspunsurilor la orice fel de întrebări. Dacă C nu reușește, atunci mașina poate fi considerată inteligentă. Deci o mașină se comportă inteligent atunci când reușește să lase asupra utilizatorului impresia că partenerul lui nu este o mașină¹. Până la ora actuală nici un program nu a reușit să treacă testul Turing.

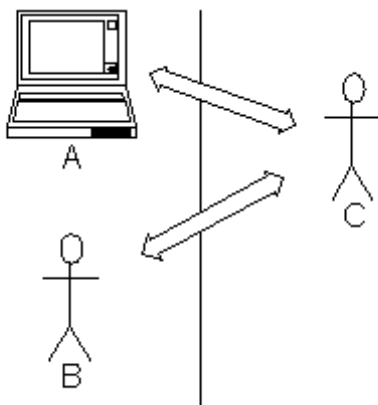


Fig. 2 Testul Turing

Cu toate acestea, testul Turing sugerează o măsură a performanțelor unui program dacă se consideră domenii restrânse ale discursului. De exemplu, un program de șah poate ajunge să aibă performanțe similare cu cele ale unui maestru, iar oponentul să nu știe dacă joacă șah cu o persoană sau cu un program. Într-o partidă de șah la distanță este greu să-ți dai seama dacă partenerul este om sau mașină. Oare poate fi considerat sistemul *Deep Blue* creat în 1997 de IBM, care l-a învins pe maestrul de șah Gari Kasparov, ca fiind inteligent? Din studiile efectuate, a reieșit că în creierul uman se află

¹ [Turing 50] Turing A.M. -Computing Machinery and Intelligence, Oxford University Press, vol LIX, no. 236, 1950.

aproximativ 100 miliarde de neuroni, fiecare fiind capabil de 1000 de operații pe secundă. În jur de 30 de miliarde de neuroni formează "materia cenușie", cea care gândește, restul de 70 de miliarde constituind "materia albă", cea care face legătura între neuronii din "materia cenușie". În contrast, Deep Blue conținea 480 de procesoare create special pentru jocul de șah, fiecare dintre ele fiind capabil să gândească aproximativ 2 milioane de poziții pe secundă.

Testul Turing simbolizează idealul pe termen lung al inteligenței artificiale ca ramură a informaticii. Turing considera că cel mai bun drum către realizarea unei mașini care să treacă testul său nu este programarea unui calculator dotat cu o mulțime fixă de cunoștințe, ci, mai degrabă, educarea unei mașini-copil, capabilă să învețe din experiență și să folosească limbajul natural ca să-și îmbogățească cunoștințele. Ea ar putea să-și rezolve problemele proprii și să-și împlinească propriile sale planuri, dând dovadă de inteligență practică în viața de zi cu zi.

Sistemele actuale de inteligență artificială conțin cunoștințe ce sunt înmagazinate de calculator și pot fi accesate atât de către calculator cât și de către utilizator. Din păcate nu există o idee consistentă referitoare la modul de reprezentare al cunoașterii. Nu avem o teorie atotcuprinzătoare despre o asemenea cunoaștere, cum ar fi cea bazată pe ideile, gândurile unor mari personalități, ca de exemplu relativitatea pentru Einstein, gravitația pentru Newton sau gazele pentru Boyle. Există două opinii despre stocarea cunoașterii care stau față în față. În primul rând sunt "eleganții", care consideră că inteligența artificială are principii fundamentale structurate, generale, sistematizabile, formale, testabile, ce utilizează logica deductivă. Purtătorii de cuvânt ai acestora sprijină limbajul de nivel înalt PROLOG, care se clădește prin declarații, stochează relațiile dintre obiecte și trage concluzii din ele.

Față în față cu eleganții stau "lejerii", după părerea cărora inteligența umană funcționează după reguli ad-hoc sau euristice, mult mai lejere, are mai puține reguli fundamentale structurate (sau acestea sunt chiar inexistente). Abordarea lor se materializează în LISP, care este un limbaj destinat în special pentru gestionarea listelor și tratarea simbolurilor. LISP domină mai ales în SUA pe când PROLOG este agreat în primul rând în Europa. Întrucât cunoștințele noastre referitoare la om, la sistemul lui de înmagazinare a cunoștințelor și inteligența lui nu sunt complete, probabil această dispută nu poate fi încă soluționată.

Între timp, în domeniul inteligenței artificiale oamenii lucrează, în majoritatea cazurilor, la reguli, eventual clădesc sisteme expert fundamentate pe euristică. În răstimpul în care cercetătorii au pus accentul pe gândire și inteligență, s-au cercetat și alte proiecții ale inteligenței artificiale. S-a acordat atenție dezvoltării roboților. A renăscut visul original de a realiza suplinitorul electronic, mecanic complet al omului. Această idee a fost neglijată temporar în interesul roboților mici. Aceștia sudează componentele caroseriilor la automobile, fierb ceai sau cafea, ori dezamorsează bombe. Până ca roboții să deprindă toate iscusințele omului, ei își continuă munca precisă, repetitivă (fără pauze de cafea, concedii sau greve).

2.2. OBIECTIVELE CERCETĂRII ÎN DOMENIUL INTELIGENȚEI ARTIFICIALE

Termenul de *Inteligență Artificială (IA)* se bucură de o tot mai largă circulație. Aproape că nu mai există domeniu care să nu aibă tangențe cu această nouă tehnologie. Inteligența artificială este folosită în colectarea de informații, luarea de decizii, rezolvarea problemelor, educație, monitorizarea și automatizarea activităților, planificare, design, comunicații etc.

Inteligența artificială a fost definită în diverse moduri. Astfel, Barr și Feigenbaum² o privesc ca un domeniu al științei calculatorului care are ca obiectiv conceperea sistemelor de calcul inteligente, inteligența fiind "un amalgam de talente de reprezentare a informației și de prelucrare a acesteia".

P. Winston, în "Artificial Intelligence"³, definește IA ca domeniul care se ocupă cu studiul ideilor care permit calculatoarelor să devină inteligente.

M. Drăgănescu, în "A doua revoluție industrială: microelectronica, automatica, electronica - factori determinanți"⁴, o definește ca proprietate a structurilor informaționale dinamice, constituite din simboluri, indiferent de forma și suportul lor fizic, implicând ordonarea sistemică a simbolurilor și o semantică de ordinul I, caracterizată prin recepționarea de informații externe ei, pe plan pur informațional sau în raport cu o realitate materială (substanțială), prin construirea de metode proprii și cunoaștere asupra acesteia, inclusiv de comunicare cu aceasta.

Hunt, în "Artificial Intelligence"⁵, subliniază că este dificil de identificat principiile unificatoare care să permită o bună definire a domeniului inteligenței artificiale. Problema care se pune este legată de definirea conceptului de inteligență naturală.

Peter Collin⁶ definește inteligența artificială drept "designul și instrumentarea aplicațiilor pe calculator care își propun să imite inteligența umană și funcțiile de luare a deciziilor, realizând raționamente fundamentale și alte caracteristici umane".

O definiție mai detaliată se găsește în Encyclopedia lui Stuart Shapiro⁷: "Inteligența artificială este studiul căilor prin care computerele pot fi făcute să rezolve sarcini cognitive la care, în prezent, omul este încă mai performant. Problemele care cad sub incidența inteligenței artificiale includ sarcini obișnuite cum ar fi înțelegerea limbajului natural, căutarea unui obiect care se află undeva mai departe etc. În plus, ea cuprinde și sarcini de expert, cum ar fi: diagnoza medicală, designul unor sisteme computaționale, descoperirea unor zăcămintele minerale, planificarea experimentelor științifice etc".

Granițele cercetării inteligenței artificiale sunt greu de trasat. Însăși definiția științifică a noțiunii de inteligență este încă deschisă, nerezolvată. Nu există o asemenea "substanță" care să se numească inteligență. Într-una din lucrările sale⁸, Sloman afirma: "există numeroase "priceperi", calități pentru a vedea diverse lucruri, pentru înțelegerea limbii, pentru planificare și testare, pentru rezolvarea problemelor, pentru conducerea activităților noastre și urmărirea lor". Toate aceste calități presupun cunoștințe specifice unor domenii pe care oamenii le folosesc (de cele mai multe ori inconștient) în cadrul unor activități profesionale sau în viața cotidiană. Definirea explicită și depozitarea acestor cunoștințe este una din realizările cercetărilor în domeniul inteligenței artificiale care au condus la apariția bazelor de cunoștințe și a sistemelor expert. Într-o bază de cunoștințe se găsesc, pe lângă faptele și noțiunile din domeniu și cunoștințe de tip *know-how*, experiența, dar și manifestările proceselor intuitive de gândire ale

² [Barr86] Barr A., Feigenbaum E.A., "The Handbook of Artificial Intelligence", volumes 1-4, Addison-Wesley, Reading, MA, 1986.

³ [Winston92] Winston P.H. – "Artificial Intelligence" (3rd ed.) Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1992

⁴ [Drăgănescu80] Drăgănescu M – "A doua revoluție industrială: microelectronica, automatica, electronica - factori determinanți"; Editura Tehnică, București, 1980.

⁵ [Hunt75] Hunt E.B., "Artificial Intelligence", New York, Academic Press, 1975.

⁶ [Collin88] Collin P. – "Dictionary of Computing", First Edition, Peter Collin Publishing, Teddington, 1988

⁷ [Shapiro87] Shapiro S. -Encyclopedia of Artificial Intelligence. Ed. Stuart C. Wiley - Interscience Publication, vol. 1, 1987.

⁸ [Sloman78] Sloman A –The Computer Revolution in Philosophy (Harvester Studies in Cognitive Science), The Harvester Press, Ltd, 1978

specialistului care-și exercită meseria la un nivel calitativ înalt. Din punctul de vedere al posibilităților de înmagazinare în sistem, putem considera orice capacități ale ființei umane, orice componentă a inteligenței naturale, astfel:

- Capacitatea surprinderii senzațiilor subiective, sesizarea (auz artificial, vedere artificială)
- Capacitatea de a gândi la ceva, atenția
- Capacitatea sesizării faptelor necăutate de la început, observația
- Capacitatea de a învăța, autoperfecționarea
- Capacitatea de cumpănire, decizie, planificare, acțiune, inclusiv capacitatea de comparare a obiectivelor
- Capacitatea de a realiza și manevra simboluri, modele, reprezentări
- Capacitatea de gândire propriu-zisă, raționarea (elaborarea de concluzii)

Cercetarea în domeniul inteligenței artificiale are trei mari obiective⁹:

- *Teoretice* - analiza explicațiilor posibile și aplicabile în practică referitoare la înțelegerea comportării raționale
- *Experimentale* - studierea experimentală și modelarea sistemelor inteligente existente (în special a capacităților umane)
- *Practice* - realizarea unor produse inteligente

Principalele domenii de utilizare ale Inteligenței Artificiale¹⁰ sunt ilustrate în figura 3:

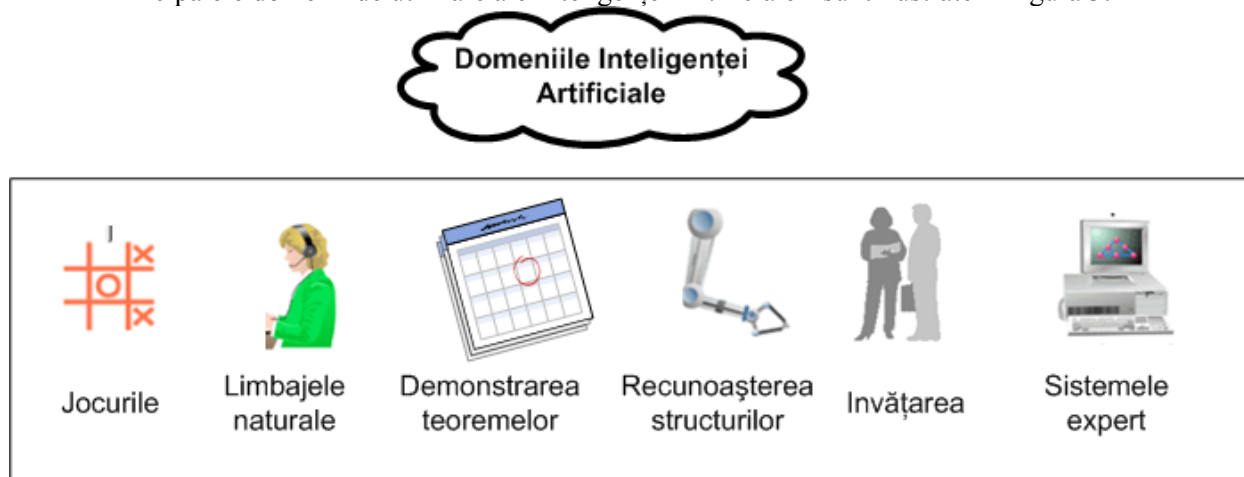


Fig.3 Domeniile Inteligenței Artificiale

Jocurile -Fundamentele teoretice se regăsesc într-un studiu al lui Shannon¹¹; în 1958 Alex Brenstein reușește să ruleze pe un IBM 704 un program bazat pe aceste idei. Numărul programelor de șah crește, precum și forța lor, astfel că nu este de mirare că se organizează campionate naționale și chiar mondiale, primul dintre acestea fiind câștigate de programul rusesc, purtând numele zeiței șahului, Caissa. Nici alte jocuri nu au fost uitate: A.L. Samuels¹² se ocupă de jocul de dame, alți specialiști¹³ de GOMOKU și exemplele ar putea continua.

⁹ [Sloman78] Sloman A -The Computer Revolution in Philosophy (Harvester Studies in Cognitive Science), The Harvester Press, Ltd, 1978

¹⁰ [Griffith87] Griffith M., Palissier C. -Algorithmic Methods for Artificial Intelligence. Kogan Page, 1987.

¹¹ [Shannon50] Shannon C. E. -Programming a Computer for Playing Chess. Philosophical Magazine, 7, 1950.

¹² [Samuels59] Samuels A. L. -Some Studies in Machine Learning Using the Game of Checkers. IMB Journal of Research and Development, 3, 1959

¹³ [Elcock67]Elcock E.W., Murray A.M. -Experiments with a Learning Component in a GO-MOKU Playing Program. In: Collins, N.L. & Michie, D. (Ed.), Machine Intelligence 1, Oliver and Boyd, Edinburgh, 1967.

Limbajele naturale -Deși la începutul anilor '60 se prefigura o soartă tristă pentru traducători și interpreți, în anii '70 traducerea automată mai era încă un vis irealizabil¹⁴. O a doua fațetă a acestui domeniu o reprezintă utilizarea limbajelor naturale pentru comunicarea cu computerul. Prima utilizare a limbii engleze ca interfață de comunicare este menționată de către McCarthy în 1958, dar se pare că programul ELIZA¹⁵ a fost primul care a putut purta o conversație.

Demonstrarea teoremelor -Mecanizarea demonstrării teoremelor cu ajutorul computerelor a făcut primii pași încă în anii '50^{16, 17}, dar au fost necesare aproape două decenii pentru a se descoperi că scrierea unui program este analoagă cu demonstrarea unei teoreme¹⁸.

Recunoașterea structurilor -Este un domeniu important, cu un număr crescând de aplicații, cele mai răspândite ținând de recunoașterea vocii (pentru controlul mașinilor), identificarea vizuală a obiectelor (pentru manipularea lor de către roboți) sortarea (clasificarea) celulelor (în analizele necesare diagnozei), recunoașterea caracterelor unui text (citire optică).

Învățarea -Termenul are două semnificații: prima se referă la procesul care are loc atunci când un program profită de experiență (se autoperfecționează), iar cea de a doua se referă la întreaga gamă de posibilități de utilizare în domeniul învățării umane, cea mai cunoscută fiind învățarea asistată de calculator.

Sistemele expert -Reprezintă una dintre aplicațiile mai noi ale inteligenței artificiale, termenul intrând în uz la finele anilor '70. Ele constituie o clasă particulară de sisteme informatice bazate pe inteligență artificială, având drept scop reproducerea cu ajutorul calculatorului a cunoștințelor și raționamentelor experților umani. Ele reprezintă deci un ansamblu de aplicații care, într-un domeniu de specialitate foarte bine delimitat, oferă servicii speciale de expert (consultanță), rezolvă probleme complexe independent sau în colaborare strânsă cu utilizatorul său. În ciuda propagandei comerciale, ele nu sunt înlocuitorii magici, atotcuprinzători ai omului, ci pachete de programe rulate pe calculatoare, având limitele și avantajele lor asemănător cu alte pachete de programe. Răspunsurile și întrebările oferite de sisteme expert sunt întotdeauna comprehensibile, accesibile și relativ ieftine.

Un sistem expert este construit pentru a oferi răspunsuri la problemele unui anumit domeniu, răspunsuri care ar trebui să fie la nivelul calitativ al unor experți umani. Abilitatea de a învăța, conform specialiștilor, aparține și ea în majoritate conceptului de sistem expert, reprezentând astfel o a doua calitate importantă prin care cercetarea inteligenței artificiale a condus la realizarea acestor sisteme. Printre sistemele expert existente astăzi, sunt unele care dispun de o abilitate pasivă de a învăța. Acestea pot acumula cunoștințe, reguli noi, prin reprogramare sau restructurare de multe ori profundă. În realizarea abilității de învățare, deocamdată se cunosc doar rezultate incipiente. Din acestea se poate observa deja că, pentru învățare (folosind experiența), sistemele au nevoie de o cantitate importantă de antecunoștințe specifice domeniului și nu numai.

O componentă importantă a sistemelor expert o reprezintă sistemele de raționare (mecanismele de inferență). Acestea sunt dezvoltate și folosite în principal pentru verificarea relațiilor logice. Există aplicații în care acestea sunt concepute pentru demonstrații matematice și logice, validări (de exemplu la configurarea resurselor hardware/software) fără a fi sisteme expert propriu-zise. Din ce în ce mai multe firme oferă suport "on-line" consumatorilor prin intermediul unor baze de cunoștințe (actualizate oportun) combinate cu mecanisme de inferență și interfețe prietenoase, evitând astfel factorii aleatorii ce derivă din natura umană a unor angajați pe asemenea posturi.

De dezvoltarea sistemelor expert se leagă și alte domenii din cadrul inteligenței artificiale. În primul rând ar fi vorba de prelucrarea limbajelor naturale, precum și recunoașterea și sinteza vorbirii. Cercetarea acestora s-a născut din dorința utilizatorilor de a comunica prin viu grai cu mașinile. Deși

¹⁴ [Kent61] Kent A. -Information Retrieval and Mechanical Translation. Vol. 1-2, Interscience Publishers, 1961

¹⁵ [Weizenbaum66] Weizenbaum J.L. -A Computer Program for the Study of Natural Language Communication between Man and Machine. CACM 0, 1, 1966.

¹⁶ [Davis57] Davis M. -A Computer Program for Pressburger's Procedure. Summer Institute for Symbolic Logic, 1957.

¹⁷ [Newell57] Newell A., Shaw J.C., Simon H. -Empirical Explorations with the Logical Theory Machine. Proceedings of WJCC, 1957

¹⁸ [Griffith87] Griffith M., Palissier C. -Algorithmic Methods for Artificial Intelligence. Kogan Page, 1987.

există câteva rezultate remarcabile în acest domeniu, ele nu sunt încă destul de răspândite. Astfel, în majoritatea sistemelor de inteligență artificială, dialogul se desfășoară cu ajutorul monitorului și tastaturii.

Printre perspectivele nu foarte îndepărtate ale acestor cercetări putem menționa și aplicarea rezultatelor sistemelor expert în sistemele de dispecerat, în domeniile traducerilor, a învățării asistate de calculator. Asemenea facilități vor conduce cu siguranță la creșterea eficienței sistemelor expert.

2.3. INTRODUCERE ÎN DOMENIUL PIETEI DE CAPITAL

2.3.1. Inteligența artificială în economie

Pe fundalul trecerii de la era informației la cea a comunicației, schimbările profunde ce au revoluționat această lume se înscriu firesc în mișcarea vastă ce afectează toate sectoarele de activitate. Dezvoltarea într-un ritm exponențial a cunoașterii umane, diversificarea acesteia, fac din ce în ce mai dificilă stăpânirea ei și aproape imposibilă aplicarea acestei cunoașteri în timp real. Inteligența artificială nu doar ajută la înmagazinarea acestei cunoașteri ci încearcă să transforme calculatorul dintr-un instrument într-un colaborator, capabil de procese inferențiale similare celor umane.

Este din ce în ce mai greu de găsit domenii în care inteligența artificială nu a pătruns sau nu poate pătrunde în viitorul apropiat, deoarece ea poate fi folosită oriunde eficiența poate fi sporită prin tehnologiile și instrumentele pe care le oferă.

Domeniul economic beneficiază din plin de cercetările din inteligența artificială. Sfera largă de aplicații de acest gen existente în management, marketing, contabilitate, finanțe și bănci impun din ce în ce mai mult metodele și tehnicile inteligente ca și instrumente indispensabile în activitatea economică.

Având în vedere că cercetarea mea se va axa pe studiul sistemelor de asistare a deciziilor și a sistemelor expert aplicate îndeosebi în domeniul bursier, încercăm în continuare să schițăm câteva aspecte din problematica pieței de capital, componentă a domeniului financiar-bancar.

2.3.2. Aspecte privind domeniul financiar -bancar

Sistemul financiar-bancar reprezintă un domeniu de mare importanță în desfășurarea operațiunilor economice și financiare într-o economie de piață. Acest domeniu s-a dezvoltat puternic susținut de tehnologia modernă, de progresele informaticii și telecomunicațiilor.

*Domeniul financiar-bancar*¹⁹ este un sistem integrat și specific, deschis și dinamic format din următoarele componente:

- instituții și organisme specifice
- cadrul legislativ adecvat
- metode, instrumente, tehnici, mijloace, concepte și modele utilizate efectiv
- sisteme informaționale și/sau informatice
- personal de specialitate
- relații financiar-bancare interne și internaționale
- conexiuni stabilite în scopul derulării unor operațiuni/tranzacții active/pasive
- sisteme de tehnică de calcul și telecomunicație utilizate.

Instituțiile și organismele specifice acestui domeniu sunt:

- Ministerul Finanțelor
- Ministerele economice
- Curtea de Conturi
- Garda Financiară
- Direcția Generală a Finanțelor Publice și Controlul Financiar de Stat
- Administrațiile financiare ale județelor și sectoarelor municipiului București
- Banca Națională a României

¹⁹ [Davidescu98] Davidescu D. N. și alții –Sisteme informatice financiar-bancare, Editura All Beck, București, 1998

- Societățile Bancare
- Casa de Economii și Consemnațiuni
- Bursa de Valori
- Casele de schimb valutar
- Societățile de asigurare-reasigurare
- Fondurile mutuale
- Societățile de investiții
- Agenții economici
- Organismele financiar-bancare internaționale cu sedii în România

Sistemul financiar-bancar îndeplinește o serie de funcții dintre care cele mai importante se referă la²⁰:

- *Efectuarea plăților* –asigură cadrul efectiv pentru executarea plății bunurilor și serviciilor
- *Atragerea economiilor* persoanelor fizice și juridice –asigură colectarea acestor economii în vederea plasării lor
- *Acordarea de credite* -asigurarea cadrului necesar încurajării economiilor va duce la crearea unor disponibilități ce pot fi folosite în zone productive și cu un coeficient de risc cât mai scăzut. Astfel este facilitat transferul de fonduri de la cei ce înregistrează excedente către cei care se confruntă cu necesități financiare.
- *Colectarea impozitelor și taxelor și efectuarea cheltuielilor bugetare* –permite Guvernului să obțină impozitele și fondurile de care are nevoie pentru îndeplinirea obiectivelor atât de funcționare a instituțiilor bugetare cât și pentru sprijinirea agenților economici în vederea atingerii obiectivelor de creștere economică
- *Asigurarea mobilității capitalurilor* prin trecerea capitalului din domeniile mai puțin eficiente în cele cu eficiență mai mare

Un sistem financiar-bancar eficient și solid, care beneficiază de încrederea tuturor participanților (agenți economici, instituții, investitori etc) reprezintă piatra de temelie a oricărei economii. Economia ar stagna dacă nu s-ar putea baza pe mobilitatea fondurilor, pe fluxul continuu de intrări și ieșiri de fonduri, reprezentând atragerea și distribuirea fondurilor. Aceste fluxuri au la bază banii, permițând încheierea tranzacțiilor de vânzare și cumpărare, evaluarea bunurilor și serviciilor, distribuirea resurselor etc.

În cadrul sistemului financiar-bancar există două tipuri de circulație financiară:

- *Directă* –concretizată prin schimbul direct între cel ce acordă și cel ce face un împrumut. Acest tip de transfer de fonduri este nesemnificativ într-o economie globală și ineficient pentru organizații ce sunt nevoite să apeleze la o multitudine de investitori individuali
- *Indirectă* –este realizată prin instituții specializate, respectiv prin intermediari financiari, care reunesc capitalurile investitorilor și le pun la dispoziția celor ce doresc să le utilizeze.

2.3.3. Locul și rolul piețelor de capital în cadrul piețelor financiare

Orice economie națională se caracterizează prin existența și funcționarea unor piețe specializate, unde se întâlnesc și se reglează cererea și oferta de active financiare și capitaluri, necesare creării de resurse pentru dezvoltarea producției de bunuri și servicii. Aceste tranzacții se realizează prin intermediul *piețelor financiare*²¹.

Piața financiară este formată din două mari sectoare, constituite ca piețe distincte:

²⁰ [Fodi99] Fodi A., Constantinescu L., Postăvaru N –Introducere în piața de capital și bursa de valori, Editura Matrix Rom, București, 1999.

²¹ [Ghilic97] Ghilic-Micu B. -Bursa de Valori, Editura Economică, 1997.

- *piața monetară* –aceasta este specializată în efectuarea de tranzacții cu active monetare, cu scadențe relativ scurte (sub 1 an), incluzând depozitele bancare la vedere și la termen, cambii, bilete la ordin, cecuri, certificate de depozit și alte titluri mobiliare pe termen scurt.

- *piața de capital* - asigură întâlnirea dintre cererea și oferta de capital, pe orice termen, prin intermediul titlurilor de creanță negociabile. Piața de capital are la rândul ei două compartimente: piața primară (piața emisiunilor de titluri) și piața secundară (piața pe care se negociază titlurile deja emise). Pe piața primară se efectuează plasamente numai în momentul creării (emiterii) titlurilor de către agenții economici, publici sau privați, prin intermediul băncilor. Excepția o constituie operarea majorării capitalului social al unei societăți a căror acțiuni sunt deja cotate pe una din piețele bursiere²². Piața secundară răspunde necesității existenței unei piețe organizate pentru deținătorii de titluri care doresc să-și recupereze fondurile fără să aștepte scadența (obligațiunile au în general scadențe lungi, iar acțiunile nu au scadență, durata lor fiind egală cu aceea a societății emitente). Astfel, orice investitor poate oricând să-și recupereze capitalul investit, transferând riscul asupra celui ce vrea să și-l asume.

În literatură se folosesc noțiuni diferite pentru a descrie structura pieței. Astfel, în unele lucrări americane piața de capital, componentă a pieței financiare, desemnează atât tranzacțiile cu titluri pe termen lung cât și pe cele cu instrumente monetare (*money market instrument*; certificate de depozit, bonuri de tezaur, etc). În literatura franceză noțiunii de piață financiară îi corespunde cea de piață de capital, aceasta cuprinzând piața creditului, piața valorilor mobiliare (numită în acest caz piață financiară) și piața monetară.

Procesul de privatizare desfășurat la noi în țară a dus la apariția și dezvoltarea piețelor de capital pe care se realizează emisiunea și negocierea titlurilor. La Bursa de Valori se poate face negocierea titlurilor societăților admise la cotaie, societăți ce îndeplinesc anumite condiții impuse de Regulamentele Bursei de Valori. Pentru titlurile societăților ce nu îndeplinesc condițiile admiterii la cotaie s-au creat piețe noi, unde accesul este permis în condiții mai puțin limitative. Una dintre aceste piețe este piața O.T.C. (*Over the Counter* - piață “la ghișeu”) sau piața interdealeri.

Piața O.T.C. este constituită, în esență, din relațiile stabilite între comercianții de titluri (dealeri), precum și dintre aceștia și clienții lor, astfel încât are semnificația unui ansamblu de tranzacții²³. Ea se caracterizează printr-o lipsă a localizării, tranzacțiile realizându-se în cadrul societăților de valori mobiliare, prin intermediul unei rețele de comunicații electronice între dealeri. Reglementările sunt mai puțin ferme și sunt făcute de regulă de asociațiile dealerilor (la noi de către Asociația Națională a Societăților de Valori Mobiliare) fiind apoi avizate de Comisia Națională a Valorilor Mobiliare.

În România, piața O.T.C. folosește sistemul de tranzacționare R.A.S.D.A.Q. (*Romanian Association of Securities Dealers Automated Quotations System*). Pe piața O.T.C. nu există un preț unic, astfel că prețul aceleiași valori mobiliare poate să difere de la un dealer la altul. Negocierea pe piața O.T.C. se face cu un anumit număr de dealeri numiți “market maker”-i (creatori sau formatori de piață) ce acționează în cadrul tranzacției în calitate de vânzători sau cumpărători. Concurența între market-maker-i impune folosirea unui sistem computerizat prin care fiecare dealer încearcă să-și atragă clienți folosind informațiile de pe piață urmând ca pe baza acestora să-și stabilească propria lor listă de prețuri aferente valorilor mobiliare. În prezent la noi în țară se desfășoară procesul de fuziune dintre Bursa de Valori și piața Rasdaq, ca măsură de consolidare a pieței de capital și de sporire a atractivității acesteia.

În figura 4 încercăm să schițăm un model al pieței bursiere, model ce cuprinde entitățile care activează pe această piață și legăturile care există între acestea.

²² [Ciobanu97] –Bursele de valori și tranzacțiile la bursă, Editura Economică, București, 1997.

²³ [Popa93] Popa I -Bursa, vol I, Colecția Bursa, București, 1993.

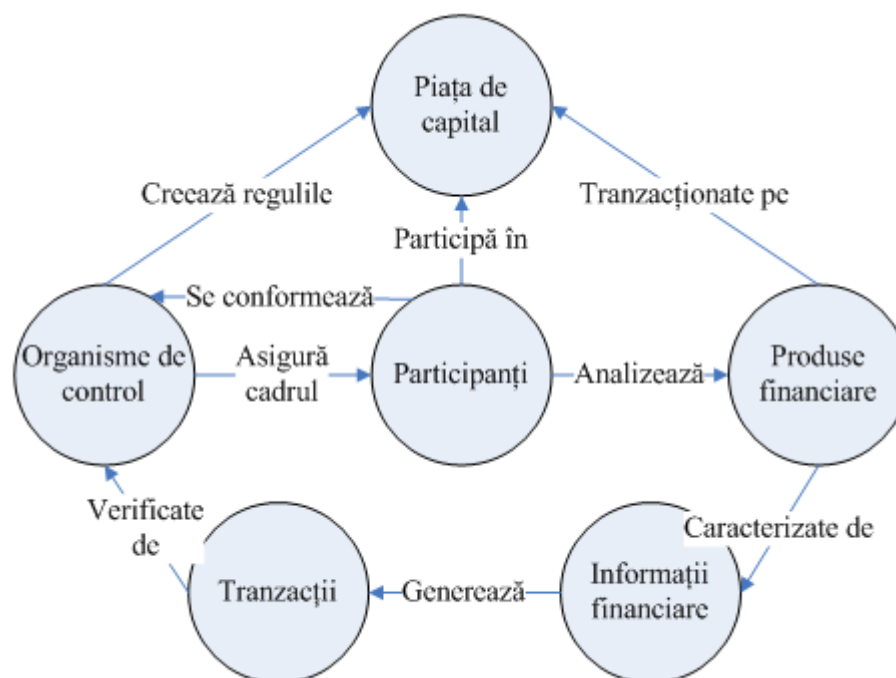


Fig. 4 Modelul pieței bursiere²⁴.

Formarea și dezvoltarea pieței capitalurilor în România reprezintă o componentă esențială a procesului de restructurare a sistemului economic în țara noastră și de creare a mecanismelor și instituțiilor specifice unei economii de piață. Din punct de vedere fundamental, România prezintă încă mari semne de întrebare pentru capitalurile speculative internaționale care pot lichidiza piața, iar unul dintre motive este faptul că piața de capital nu este pregătită să facă față la acestea, pe de o parte datorită faptului că majoritatea societăților se află încă în proces de restructurare, iar pe de altă parte investitorii nu au încă experiența necesară să reacționeze corect la aceste mișcări. Principalele direcții de perfecționare a pieței de capital pot fi evidențiate prin dezvoltarea tranzacțiilor cu obligațiuni și prin implementarea titlurilor financiare derivate, contractele futures și opțiunile putând deveni obiectul principal al speculațiilor bursiere.

2.3.4. Politici investitoriale

Valorile mobiliare sunt documente sau titluri de valoare care dau drepturi de participație sau creanță și sunt negociabile pe piața secundară. Ele se vând și se cumpără de către participanții la piața de capital, asigurându-se un flux permanent între investitori (oferanții de capital) și utilizatori (agenți economici în căutare de capital). Investițiile în valori mobiliare sunt făcute de clienți din diferite motive. Indiferent de motivul acestor investiții, investitorul urmărește două obiective: creșterea capitalului și a venitului adus de acesta. Există trei tipuri de politici pentru portofoliile de investiții²⁵:

- *politica defensivă* - are ca obiectiv conservarea capitalului și reducerea riscului. Un astfel de portofoliu se constituie în general din obligațiuni care aduc un venit fix și sunt garantate fie de stat fie prin activele întreprinderilor. Venitul adus este în acest caz moderat dar sigur.

- *politica agresivă* - scopul principal al acestei politici îl reprezintă maximizarea câștigurilor. Portofoliul în acest caz se constituie în principal din acțiuni care sunt riscante dar asigură un grad înalt de valorificare a capitalului.

²⁴ adaptare după [Freedmann93] Fredmann R. –AI on Wall Street –The Encyclopedia of Computer Technology, Volume 28, Marcel Dekker Inc., New York, 1993.

²⁵[Intrados95] XXX -Manual de pregătire a brokerilor -elaborat de ANSVM-Grupul Intrados, 1995.

-*politica agresivă/defensivă* - este o politică echilibrată, asigurându-se un echilibru între acțiuni și obligațiuni.

În cursul perioadelor de expansiune economică se recomandă cumpărarea de acțiuni în timp ce în perioadele de recesiune economică se recomandă achiziționarea de titluri cu venit fix (obligațiuni).

Domeniile de activitate, din punctul de vedere al investițiilor, se împart în trei categorii²⁶:

-*domenii de activitate defensive* - includ rate ale profitului moderate, constante și sigure datorită specificului activității (domeniul alimentar sau cel al medicamentelor).

-*domenii de activitate în expansiune* - se caracterizează prin rate ale profitului mari, în creștere, dar profiturile sunt reinvestite ceea ce duce la dividende mici (industria calculatoarelor și industria electronicii).

-*domenii de activitate ciclice* - caracteristic pentru aceste domenii este faptul că prețul acțiunilor, dividendele, activitatea în general, sunt ciclice (domeniul construcțiilor și industria automobilelor) deci se caracterizează printr-o fluctuație mare a vânzărilor, a beneficiilor și a dividendelor.

2.3.5. Caracteristicile investițiilor în valori mobiliare

Problema cu care se confruntă mulți investitori este aceea dacă o investiție este sau nu potrivită pentru ei. Pentru a rezolva această situație trebuie să se ia în considerare următorii factori²⁷:

1. Venitul - reprezintă suma de încasat reală sau anticipată ce va fi realizată dintr-o investiție (în cazul acțiunilor aceasta constă din dividende și din creșterea cursului acțiunilor, în cazul obligațiunilor constă din dobândă și din diferența dintre valoarea nominală și prețul de achiziție pentru obligațiunile cu discount sau dintre valoarea de rambursare și valoarea nominală).

2. Costurile - reprezintă comisioanele aferente tranzacțiilor. Unele investiții necesită și plata unor taxe suplimentare (cum ar fi ipotecarea proprietăților).

3. Efectul de levier - reprezintă efectul de multiplicare a profitului (pierderii) pe unitate de valoare investită. De exemplu, tranzacțiile la termen necesită doar o acoperire parțială (marja), și în consecință se multiplică cu atât mai mult profiturile (pierderile) în cazul unei previziuni corecte (respectiv incorecte) cu cât acoperirea este mai mică. În acest caz suma de bani cu care se cumpără sau se vinde un contract futures reprezintă doar un mic procent (de regulă maxim 10%) din valoarea activului suport care constituie obiectul tranzacției.

4. Protecția împotriva inflației - presupune că pentru a afla rata reală a rentabilității trebuie ca din rata rentabilității să se scadă rata inflației.

5. Riscul - Prin riscul unui titlu mobilier se înțelege fie sacrificiul unui avantaj imediat sau absența unui consum imediat, în schimbul unor avantaje viitoare, fie incertitudinea asupra valorii unui bun financiar înregistrată la o dată viitoare. Unele investiții sunt relativ lipsite de riscul nerecuperării investiției (obligațiunile guvernamentale) în timp ce altele au un grad ridicat de risc (obligațiunile precare sau murdare - "junk bonds", emise de societăți ce n-au lichiditatea asigurată).

Plasamentele se efectuează pe baza situației bursiere a diferitelor titluri la un moment dat. Situația titlurilor este într-o permanentă schimbare, de aceea investitorul trebuie să urmărească permanent evoluția cursurilor titlurilor din portofoliul său. Pentru protejarea unui portofoliu de titluri împotriva variațiilor cursurilor titlurilor, investitorul trebuie să dispună de o structură diversificată a portofoliului ("Nu pune toate ouăle într-un singur coș"), astfel încât creșterile de curs ale unor titluri vor compensa scăderile de curs ale altor titluri. Se recomandă în acest sens deținerea atât de acțiuni din cel puțin șapte domenii de activitate diferite (până la 30, în funcție de piață) cât și de obligațiuni. Această structură trebuie schimbată în funcție de evoluția economico-financiară a societăților emitente. Stabilirea unei structuri a portofoliului cât mai avantajoasă presupune aprecierea corectă a titlurilor care se pretează cel mai bine obiectivelor urmărite.

Alegerea momentului investiției este la fel de importantă ca strategia ca și alegerea acțiunilor. Dacă pentru un investitor pe termen lung aceasta nu are o importanță capitală, pentru un investitor pe

²⁶ idem

²⁷ [Intrados95] XXX -Manual de pregătire a brokerilor -elaborat de ANSVM-Grupul Intrados, 1995.

termen scurt alegerea acestui moment înseamnă profit sau pierdere. În general se recomandă ca acțiunile să se vândă și să se cumpere la intervale regulate ale unui ciclu de piață (trebuie să se ia în considerare și nivelul comisioanelor).

Pentru a realiza plasamente avantajoase, investitorul trebuie să urmărească mai multe obiective²⁸:

- stabilirea factorilor economici și monetari care influențează activitatea de afaceri în ansamblul ei
- definirea factorilor care provoacă tranziția de la “boom” la recesiune în fiecare ramură de activitate
- stabilirea factorilor de ordin monetar și psihologic care generează fluctuații ale piețelor
- stabilirea pe baza unei analize a societăților și a ramurilor de activitate care oferă perspective interesante, în scopul de a sesiza pe cele a căror situație este solidă
- stabilirea cursurilor diferitelor titluri de valoare pentru a vinde sau cumpăra în perioadele de subevaluare sau supraevaluare temporară a cursurilor, provocate de condițiile tehnice conjuncturale ale pieței.

2.3.6. Concluzii

Pentru orice investitor, scopul oricărei investiții îl reprezintă plasarea profitabilă a capitalului disponibil. Alegerea căilor celor mai potrivite de economisire și investiție se poate dovedi a fi un proces dificil, având în vedere o serie de factori precum multitudinea instrumentelor disponibile, instabilitatea pe care o traversează anumite piețe, recesiunea economică, etc. În aceste condiții se pune problema alegerii unor instrumente de economisire adecvate, care să ofere o păstrare în siguranță a acestor economii, dar și multiplicarea lor în timp, devenind practic dintr-un instrument de economisire unul de investire. Multiplicarea economiilor trebuie nu numai să asigure acestora o protecție la inflație, ci și să aducă un câștig real, în condițiile unui risc acceptabil.

Pentru a putea selecta cele mai rentabile investiții pe piața de capital, trebuie să ne formăm o imagine de ansamblu a pieței financiare și a pieței de capital, să urmărim nu numai evoluții individuale ale acțiunilor ci și parametri precum stadiul economiei, mediul de afaceri și creșterea economică, rata inflației și deprecierea monedei naționale, dobânzile acordate de bănci la depozite și randamentele titlurilor de stat, performanțele și evoluția fondurilor mutuale, evoluția de ansamblu a piețelor bursiere, exprimată prin indicii bursieri. Aceste informații sunt importante în aprecierea oportunității și rentabilității investițiilor. Dacă rentabilitatea investiției în acțiuni nu depășește rata inflației, atunci în termeni reali înregistrăm o pierdere; urmărirea fondurilor mutuale este importantă pentru a aprecia eficiența investițiilor proprii în acțiuni: dacă aceste acțiuni aduc câștiguri mai mici sau relativ egale cu cele oferite de fondurile mutuale, atunci probabil strategia aplicată este inefficientă; în loc de acțiuni am fi putut plasa banii în fonduri mutuale și obține aceleași câștiguri, dar în condiții de risc mai redus și efort mai mic. Urmărirea indicilor bursieri este importantă pentru a distinge trendul general al pieței: în perioada de creștere a pieței perspectivele pentru investițiile noastre devin favorabile, în timp ce în perioadele de restrângere a pieței obținerea unor câștiguri este mai dificilă.

Consecințele căderilor unor piețe bursiere și efectul negativ provocat de acestea asupra economiilor au dus la formarea unor păreri conform cărora bursele nu sunt altceva decât o modalitate de circulație a banilor câștigați în urma activităților economice desfășurate, fără ca ele să aducă o valoare economică reală. Fostul președinte american, Theodore Roosevelt a afirmat cândva că “nu este nici o diferență morală între a paria pe cai sau la cărți și a paria la bursă”.

În pofida acestei concepții, bursa s-a dovedit în toată lumea, fără îndoială, ca unul dintre elementele vitale ale creșterii economice pe termen lung. Mai multe studii empirice au demonstrat că există o corelație între nivelul de dezvoltare al pieței bursiere al unei țări și nivelul său de creștere și dezvoltare economică. Statele cu o piață bursieră bine dezvoltată tind să aibă, în general, o mai mare creștere și dezvoltare economică decât cele cu piețe bursiere subdezvoltate.

²⁸ [Angas70] Angas L.L.P. -Placements et speculations en bourse, Petit bibliotheque, Ed. Payot, Paris, 1970, 16.

CAPITOLUL 3 ANALIZA PIEȚEI DE CAPITAL

În cadrul oricărui proces de luare a deciziilor de investiție, aspectul fundamental îl reprezintă cunoașterea unor informații corecte și actuale. Decizia de investiție are la bază o analiză investițională care pleacă de la evaluarea factorilor calitativi ai unui portofoliu și continuă cu determinarea momentelor optime de vânzare și cumpărare ale activelor financiare din care este constituit.

Primele întrebări care iau naștere în momentul în care dorim să tranzacționăm pe o piață (indiferent că este una de capital, de titluri financiare sau instrumente derivate cum sunt contractele futures și opțiunile) sunt: ce să cumpărăm, când să cumpărăm și când să vindem. Problema ar fi rezolvată dacă am cunoaște răspunsuri la întrebări de genul: “ar fi bine dacă cumpărăm astăzi? sau ar trebui să așteptăm până mâine? încotro se va mișca prețul în următoarele zile?”. Investitorii caută prin numeroase metode să găsească o modalitate de a anticipa mișcările cursurilor, o formulă de calcul a prețurilor care să le permită să fie permanent cu un pas înaintea celorlalți sau pur și simplu să elimine incertitudinile legate de fluctuațiile cursurilor respective. Unii folosesc modele matematice și statistice, alții urmăresc istoricul cursurilor în speranța descoperirii unor anumite modele standard de evoluție a prețurilor. Cercetătorii au ajuns la studiul psihologiei pieței și a participanților, însă, cel puțin până în prezent, nu s-au obținut rezultate concrete în această direcție.

Ceea ce îi preocupă pe investitori este deci posibilitatea anticipării cursurilor valorilor mobiliare. Aceste eforturi au conturat de-a lungul timpului o știință numită *analiza pieței*, iar în cadrul ei s-au conturat două curente: *fundamentaștii* și *tehnicienii*. Obiectivul lor este același: previziunea prețului, însă metodele utilizate pentru atingerea acestui obiectiv sunt diferite. Vom încerca în continuare să facem o introducere în acest domeniu prezentând elementele care stau la baza curentelor menționate mai sus, fără a avea pretenția de a acoperi întreg domeniul analizei pieței.

Teritoriu aflat prin excelență sub imperiul pragmatismului, domeniul piețelor financiare a început să-și dezvolte și o redutabilă latură teoretică. Analiza ce se face în astfel de condiții poate fi de trei tipuri:

- *analiza fundamentală*
- *analiza tehnică*
- *analiza bazată pe teoria pieței eficiente*

3.1. ANALIZA FUNDAMENTALĂ

3.1.1. Generalități

Analiza fundamentală vizează mișcarea cursurilor titlurilor pe termen mediu și lung. Aceasta nu reprezintă o simplă studiere a conturilor societății (ca analiza financiară), esența ei fiind estimarea valorii intrinseci (valoarea teoretică a acțiunilor). Valoarea intrinsecă este considerată termen de comparație pentru aprecierea realității cursului acțiunilor. Chiar dacă valoarea intrinsecă este aproximativă, abordarea ei fundamentală indică direcția de mișcare a prețului acțiunilor în viitor. Adepții analizei fundamentale se concentrează asupra valorii intrinseci a acțiunilor, ei studiază dacă actualul preț este sub sau peste valoarea intrinsecă. La analiza fundamentală se au în vedere următoarele aspecte²⁹:

- ramura de activitate de care aparține societatea emitentă
- ciclicitatea activității societății
- dacă managementul este potrivit cu activitatea societății
- gradul de dependență/independență față de factori exogeni
- strategia de viitor a firmei;
- punctele tari și punctele slabe ale societății, cum știe să-și folosească atuurile și ce face pentru eradicarea factorilor negativi din activitatea sa.

La nivelul firmei emitente a titlurilor, analiza fundamentală vizează evaluarea titlurilor societății și estimarea profiturilor societății emitente în următorii ani; la nivelul pieței de negociere a titlurilor ea are ca obiective evaluarea influenței factorilor macroeconomici și a celor de piață. Analiza fundamentală are la bază studiul comparativ al unui set de indicatori. Acești indicatori se obțin pe baza analizei rezultatelor financiare conținute în rapoartele financiare ale societății (Contul de Profit și Pierdere, Bilanțul Contabil) și ale prețului format pe baza confruntării cererii cu oferta. Pe o piață organizată, acesta din urmă va reflecta în mod justificat performanțele emitentului, prețul de piață nefiind altceva decât imaginea interesului investitorilor pentru activul financiar respectiv. Se obține astfel o evaluare a echilibrului financiar al companiei analizate.

Analistul studiază bilanțul, costurile, marja profitului, dividendele actuale și viitoare, performanțele manageriale, producția și situația financiară, perspectivele economice ale firmei, ținând cont de toate aceste caracteristici în diferitele stadii ale ciclului unui produs realizat de firmă (dezvoltare, creștere, stabilizare, declin). Se impune de asemenea studierea unor indicatori precum: masa monetară, rata inflației, rata de creștere a producției industriale, indicii bursieri, soldul comerțului exterior, etc. Considerăm în același timp că instabilitatea economică, inflația, declinul economic, situația politică și socială trebuie și ele atent studiate.

Analiza fundamentală prezintă o serie de avantaje:

- disciplinează investitorul pentru că îl obligă la analiza unor factori
 - obligă investitorul să țină seama de riscuri
 - obligă analistul să nu prognozeze valoarea intrinsecă doar pe baza PER-ului prognozat (*Price Earnings Ratio* -Raportul preț-câștig -vezi subcapitolul 3.1.2) ci să-l compare cu PER-ul din trecut și cu PER-ul altor societăți
 - presupune luarea în considerare a impactului macrofactorilor economici (cursul de schimb, inflație, șomaj, etc), social-politici, calamități naturale etc.
- În schimb, analiza fundamentală are și unele dezavantaje dintre care enumerăm următoarele:
- o anumită relativitate a previziunilor variabilelor fundamentale
 - schimbările psihologice pot avea o influență mare asupra cursului acțiunilor.

3.1.2. Indicatori folosiți în analiza cursului acțiunilor

²⁹. [Cluj96] XXX -Conferința internațională “Perspectivele bursei de valori în țările Europei Centrale și de Est”, Cluj, 22-23 martie 1996.

Analiza fundamentală a situației economico-financiare a firmelor ale căror acțiuni sau, după caz, obligațiuni se afla la dispoziția investitorilor pe piața de capital, fie că este vorba de cea primară, fie de cea secundară, se bazează pe studiul comparativ al unui set de indicatori. Acești indicatori economico-financiar sunt calculați pe baza informațiilor oferite de situațiile financiar-contabile ale firmei: bilanțul contabil, contul de profit și pierderi etc. precum și de prețul de piață al valorii mobiliare respective care reflectă interesul investitorilor pentru titlul respectiv și care este format prin confruntarea cererii cu oferta. Prin acești indicatori, analiza fundamentală își propune să determine dacă valorile mobiliare în discuție sunt supra-, sub- sau corect evaluate.

1. Market-to-book ratio -raportul valoare de piață/ valoare contabilă (patrimonială) (MBR)

Acest indicatorul arată cât se va plăti pe piață pentru o unitate a valorii nete a activelor și se calculează astfel:

$$\text{MBR} = \frac{\text{Pretul de piața pentru o acțiune}}{\text{Valoarea contabilă pentru o acțiune}}$$

unde $\text{Valoarea contabilă} = \frac{\text{Active nete}}{\text{Număr de acțiuni}}$

Indicele MBR exprimă cât de scumpă este o acțiune în raport cu valoarea contabilă a acesteia. Un preț de piață al unei acțiuni sub valoarea contabilă adică o valoare subunitară a indicatorului poate să se constituie într-un semnal de cumpărare cu condiția ca starea de sănătate financiară a firmei să fie ridicată, managementul, politica de investiții să fie orientate corespunzător și să existe piață de desfacere pentru produsele sau serviciile firmei. În general, valori peste 3-3,5 ale acestui indicator sunt considerate mari, dar și în aceste cazuri analiza trebuie făcută ținând cont de mai mulți factori.

Acest indicator este util întrucât aduce la același numitor și asigură astfel comparabilitatea mai multor acțiuni.

2. Earnings per share -profitul net pentru o acțiune (EPS)

Indicatorul arată mărimea profitului net aferent unei acțiuni. Se calculează astfel:

$$\text{EPS} = \frac{\text{Profitul net aferent acțiunilor obisnuite}}{\text{Numărul de acțiuni obisnuite emise}}$$

Un aspect pozitiv pentru acest indicator este tendința de creștere și care justifică investiția în aceste acțiuni. La scăderea valorii acestui indicator se analizează cauzele ce duc la această situație, cauze ce pot proveni fie datorită declinului respectivei societăți, fie declinului ramurii sau a pieței în general sau pot fi doar conjuncturale caz în care, în cazul investițiilor pe termen lung, vor dispărea și ca atare nu s-ar justifica vinderea acestor acțiuni.

Evoluția elementelor ce compun acest indicator este foarte importantă (se poate ca profitul net / acțiune să rămână constant datorită scăderii concomitente a profitului net și a numărului de acțiuni prin reducerea capitalului social (acest lucru nu este un aspect pozitiv pentru respectiva societate).

3. Yield sau current yield - rentabilitatea curentă a unei acțiuni (CY)

Indicatorul arată mărimea beneficiului obținut raportat la prețul de piață al acțiunii. Se calculează cu formula:

$$\text{CY} = \frac{\text{Dividend anual pentru o acțiune}}{\text{Pret curent de piața pentru o acțiune}}$$

În analiza bursieră, acest indicator este folosit doar atunci când se urmărește încasarea în viitor a dividendelor. O rată curentă în creștere de-a lungul timpului poate să conducă la ideea unei companii sănătoase. În realitate însă cuantumul dividendelor reprezintă una din politicile stabilite de firmă. Astfel, politica prin care se măresc artificial dividendele doar pentru a menține treaz interesul investitorilor nu este favorabilă intereselor de viitor ale firmei și ca atare investitorii nu privesc favorabil acest lucru deoarece afectează politica de investiții a firmei. Firmele în expansiune oferă dividende mai scăzute și ca atare rentabilitatea curentă este mai coborâtă; acest lucru nu înseamnă că respectivele acțiuni nu sunt interesante pentru investitori. În cazul acestor firme o mare parte a profitului este reinvestită ceea ce conduce la creșterea valorii firmei, iar piața reacționează prin creșterea cursului respectivelor acțiuni. O

valoare scăzută a rentabilității curente poate să însemne și un preț prea mare plătit pentru acțiunea în cauză, caz în care nu este oportună achiziția acestor titluri, recomandându-se vânzarea celor existente, deoarece este posibil ca în viitor cursul acțiunii să scadă aducându-se astfel acest indicator spre valori mai convenabile.

4. Rentabilitatea totală (RT)

Se calculează astfel:

$$RT = \frac{\text{Dividend} + P_1 - P_0}{P_0} \text{ unde:}$$

P_1 =prețul acțiunii la sfârșitul perioadei de observație

P_0 =prețul acțiunii la începutul perioadei de observație.

Acest indicator se calculează la sfârșitul anului (respectiv în momentul acordării dividendului).

O companie în plină dezvoltare are de obicei un câștig de capital ridicat dar în cazul unei rentabilități curente mici și a unui câștig de capital de asemenea mic se recomandă vânzarea acestor acțiuni (acest studiu trebuie să cuprindă mai multe perioade de observație).

Se constată prin urmare că nu există o relație clară între acest indicator și calitatea acțiunilor în cauză. O valoare ridicată a rentabilității curente este una din caracteristicile companiilor ajunse la maturitate și care au o creștere lentă. Spre achiziționarea acestor acțiuni se orientează cei ce doresc un venit curent relativ constant. Aceste acțiuni constituie investiții interesante în cazul unor rentabilități curente ridicate provenind dintr-un preț de piață coborât.

5. Price Earnings Ratio -raportul preț-câștig (PER)

Acest indicator exprimă durata de recuperare a unei investiții (prețul plătit per acțiune) din profiturile obținute de emitent (EPS). PER arată de câte ori este mai mare prețul de piață al acțiunilor față de profitul net (câștigul) aferent acestor acțiuni sau altfel spus cât de scumpă este o acțiune în raport cu profitul net ce s-ar putea să îi revină. Formula pentru calculul acestui indicator este:

$$PER = \frac{\text{Cursul acțiunii}}{\text{Profit net per acțiune (Dividend + Castig de capital)}}$$

Utilitatea principală a indicelui derivă din faptul că acesta asigură comparabilitatea acțiunilor, deoarece doar informația legată de preț este irelevantă în timp ce acest indice aduce acțiunile la un numitor comun. Cu cât indicele este mai mic (și astfel perioada de recuperare a investiției este mai redusă) cu atât investiția este mai eficientă. De regulă, indicele PER se calculează pe baza profiturilor raportate periodic de către societățile cotate. Pe măsura trecerii timpului, până la raportarea următoarelor situații financiare, valoarea indicelui scade în relevanță (un indice calculat pe baza prețului de piață din iunie și a profitului din anul precedent nu mai este relevant). În acest caz se pot utiliza diverse metode de "aducere la zi" a indicelui. Indicele PER nu se poate calcula în anii în care o firmă înregistrează pierderi.

Se consideră că valoarea raportului preț-profit net este mare când depășește 20:1 -aceasta nu este însă o regulă strictă deoarece firme în plin proces de dezvoltare pot înregistra valori ale PER-ului de 40:1 sau chiar 90:1. Firmele care au ajuns la maturitate și ca atare au o evoluție constantă înregistrează valori ale PER-ului mai mici sau în jur de 20:1 (aceste valori sunt valabile pentru piața americană). În general se recomandă o analiză a acestui indicator pe mai mulți ani, o evoluție crescătoare putând arăta o tendință de apreciere mai puternică (și mai fundamentată) decât simpla creștere a prețului respectivei acțiuni.

O valoare ridicată a acestui indicator poate sugera un optimism al investitorilor în ceea ce privește evoluția viitoare a firmei și ca atare sunt dispuși să plătească un preț relativ ridicat în ideea unui câștig viitor mai ridicat. În același timp o valoare ridicată a PER-ului poate să însemne o supraevaluare a acțiunilor firmei ceea ce este un semnal clar de vânzare sau de amânare a momentului în care să se realizeze cumpărarea. Un PER ridicat este asociat în general acțiunilor cu un grad ridicat de volatilitate (foarte sensibile la mișcările pieței pe care se negociază).

O valoare scăzută a acestui indicator (sub 10:1) se înregistrează fie la firmele ce nu se află momentan în atenția investitorilor ceea ce conduce la lipsa interesului față de acțiunile respective fie la firmele ce se află în perioade dificile și ca atare au un preț de achiziție coborât. Un PER coborât este un prim semn al deteriorării situației unei firme și ca atare se recomandă o analiză amănunțită pentru a se vedea cauzele ce au condus la această situație. În cazul în care nu se constată deficiențe în activitatea de

ansamblu a firmei, aceste acțiuni, prin prețul mic în raport cu câștigul, pot să constituie o investiție interesantă.

Indicele PER se poate calcula și la nivel sectorial sau de piață. Indicele PER al bursei de valori (calculat ca și medie a indicilor PER reprezentând toate acțiunile ce se tranzacționează sau doar a celor mai semnificative) poate arăta nivelul de sub sau supra-evaluare ai unei burse de valori. Procedura este identică și pentru indicii sectoriali, putându-se face astfel comparații între bursele de valori ale mai multor țări.

Literatura americană consideră PER-ul drept cel mai reprezentativ pentru caracterizarea eficienței investiției în acțiunile unei firme. Acest indicator poate constitui un ajutor important în luarea deciziilor de vânzare sau de cumpărare de către investitori, dar el prezintă un mare neajuns: crește puternic când profitul per acțiune tinde spre zero. De aceea, pentru evitarea concluziilor greșite, se recomandă corelarea cu alți indicatori.

6. Prices to Sales Ratio –raportul preț/vânzări (PSR)

Acest indicator reflectă raportul dintre cursul de piață al unei acțiuni și cifra de afaceri aferentă unei acțiuni.

$$\text{PSR} = \frac{\text{Cursul acțiunii}}{\text{Cifra de afaceri la o acțiune}}$$

Strategia lui Fisher indică ca niciodată să nu se achiziționeze acțiuni cu PSR mai mare decât 3 și trebuie evitate acțiunile cu un PSR mai mare decât 1,5 chiar dacă alți indicatori sunt foarte încurajatori. În general trebuie căutate acțiuni cu un PSR sub 0,75.

7. Capitalizarea de piață (CP)

Capitalizarea de piață (CP) reprezintă valoarea întregului pachet de acțiuni emis de societate, valoare calculată la prețul pieței

$$\text{CP} = \text{Numar acțiuni emise} \times \text{Pret de piata}$$

8. Return on Assets -Rentabilitatea activelor (ROA)

Acest indicator exprimă eficiența utilizării activelor, altfel spus, capacitatea managerului companiei de a obține profituri din activele pe care le are la dispoziție.

$$\text{ROA} = \frac{\text{Profit net}}{\text{Activ total}}$$

9. Return on Equity –rentabilitatea capitalurilor (ROE)

Acest indicator se calculează ca raport între profitul net și activul net al firmei:

$$\text{ROE} = \frac{\text{Profit net}}{\text{Activ net}}$$

ROE se concentrează numai asupra randamentului capitalului investit de actionari. O rată mică poate da insatisfacții acționarilor pentru ca nu dă posibilitatea distribuirii de dividende pe măsura așteptărilor și exprimă totodată o eventuală supracapitalizare. Dar o rată scăzută poate fi și un semn ca firma duce o politică de reinvestire a profitului pentru a crește activele și fluxul de venituri viitoare. Dacă multe din aceste investiții se fac în activități de cercetare-dezvoltare, care vor aduce profituri, dar nu pe termen scurt, este posibil ca în viitor ROE să crească, oferind satisfacție acționarilor. Pe de altă parte, o rată prea mare indică o subcapitalizare și deci un grad mare de îndatorare.

10.Rata de plată a dividendelor (RPD)

$$\text{RPD} = \frac{\text{Dividend anual (pentru acțiuni comune)}}{\text{Profit net per acțiune}}$$

În practica internațională, s-au structurat trei tipuri de politici de dividend, și anume:

- *politica reziduală* (oportunistă) prin care mărimea dividendului pe acțiune depinde în mare măsură de politica de finanțare și investiții a întreprinderii respective. Dacă sunt oportunități de investiții, acestea vor fi acoperite în primul rând prin autofinanțare, iar dividendul de distribuit va

fi suma din profit rămasă neinvestită. Domeniile de activitate în expansiune au un profit net per acțiune ridicat dar o rată de plată a dividendelor coborâtă, deoarece profiturile se capitalizează. Câștigul la astfel de acțiuni se constituie în special din creșterea cursului acțiunilor.

- *politica ratelor* - în conformitate cu aceasta emitentul distribuie anual un anumit procent din profitul realizat. Dividendul variază de la an la an în funcție de profitul realizat și de rata de distribuire a acestuia.
- *politica de stabilizare* sau de prudență prin care întreprinderea își propune să asigure un dividend constant sau, mai mult, o rată "g" constantă de creștere anuală a dividendului, indiferent de variațiile profitului brut.

În abordarea fundamentală problema evaluării acțiunilor presupune estimarea anumitor indicatori considerați a fi reprezentativi pentru luarea deciziilor de cumpărare/vânzare de acțiuni. Există o serie largă de indicatori ce pot fi folosiți în analiza fundamentală, dar cei prezentați mai sus sunt cei mai des utilizați.

Un aspect pe care trebuie luat în considerare este inflația. Acesta a distorsionat într-o oarecare măsură bilanțurile contabile, astfel încât pentru evoluția în timp a acestor indicatori sunt necesare anumite corecții.

Considerăm că toți acești indicatori au un rol bine definit în decizia de investiții a oricărui investitor, dar aceasta nu trebuie luată având la bază doar analiza fundamentală. Apariția situațiilor lunare sau trimestriale ale firmelor sau știrile economice importante împreună cu datele de tranzacționare furnizează "materia primă" necesară analizei fundamentale. Piața însă nu se bazează doar pe așteptări logice ci și pe influențe emoționale și de altă natură. De aceea, acești indicatori trebuie folosiți împreună cu ceea ce oferă analiza tehnică, pentru o bună fundamentare a deciziei de investiție.

3.1.3. Estimarea valorii intrinseci a unei acțiuni pe baza actualizării veniturilor viitoare

Veniturile viitoare ale unei acțiuni sunt reprezentate de dividende. Valoarea unei acțiuni se poate estima pe baza valorii actualizate a dividendelor aduse de respectiva acțiune prin relația:

$$C = D_1/(1+k_1) + D_2/(1+k_2)^2 + \dots + D_n/(1+k_n)^n$$

unde: C este valoarea estimată a acțiunii;

k_1, k_2, \dots, k_n este șirul ratelor de actualizare (poate fi rata medie a dobânzii la depozitele pe termen lung, rata dobânzii la bonurile de tezaur, etc)

D_1, D_2, \dots, D_n este șirul dividendelor pentru cele n perioade luate în calcul.

Presupunem această rată constantă de-a lungul anilor și presupunem că dividendele au o rată constantă de creștere notată g. În aceste condiții valoarea unei acțiuni se poate obține cu relația:

$$C = D/(1+k) + D(1+g)/(1+k)^2 + D(1+g)^2/(1+k)^3 + \dots$$

Astfel se ajunge la *formula lui Williams*:

$$C = D/(k-g)$$

În cazul în care $g=0$ (dividend constant de-a lungul anilor) formula devine:

$$C = D/k.$$

Considerăm că această formulă extrem de simplă trebuie aplicată cu prudență pe de o parte datorită faptului că rata de actualizare k este variabilă în timp, iar pe de altă parte deoarece e greu de presupus că rata de creștere a dividendelor g poate rămâne constantă în timp.

Dividendul reprezintă partea distribuită acționarilor din profitul net. Dacă notăm cu d fracțiunea distribuită și cu E profitul net pe acțiune, rezultă că:

$$D = d \cdot E$$

Într-o economie perfect stabilă rata de creștere a dividendelor ar putea fi zero; în realitate însă, există întotdeauna o rată a inflației pe care o notăm cu i. Pentru ca dividendul să "țină pasul" cu inflația trebuie ca $g=i$. Formula lui Williams devine în acest caz:

$$C = d \cdot E/(k-i); C/E = d/(k-i)$$

Raportul C/E reprezintă tocmai PER-ul, iar k-i reprezintă rata de actualizare afectată de inflație.

PER = $d/(k-i)$ reprezintă formula lui Kovacs.

Se poate calcula un raport între PER-ul real, observat și PER-ul teoretic astfel:

R=PER real / PER teoretic.

În perioade de scăderi ale cursurilor bursiere aceste raporturi sunt rar sub 1, în general se situează în jurul valorii de 1,2. În momentul crahului bursier ce a afectat New York-ul în octombrie 1987, acest raport depășea 1,6.

3.1.4. Analiza statistică a rentabilității și riscului valorilor mobiliare

Indicatori statistici

În general, cursul fiecărei valori mobiliare urmărește tendința pieței, tendință caracterizată printr-un indice general al pieței bursiere. Astfel se poate forma un model care să urmărească în timp relația dintre rentabilitatea unei acțiuni sau a unui portofoliu de acțiuni și tendința pieței bursiere. Astfel, una din preocupările investitorilor este de a anticipa tendințele de creștere sau scădere ale indicelui general al pieței bursiere.

Se poate defini o corelație între rentabilitate și risc conform căreia o rentabilitate mai mare este însoțită de un risc mai mare și invers. Prin riscul unui titlu mobilier se înțelege incertitudinea asupra valorii lui la o dată viitoare. Analiza riscului unui titlu mobilier face parte din teoria deciziei în condiții de incertitudine, utilizând, în cele mai multe cazuri, calculul probabilistic.

În cadrul analizei celor două variabile aleatoare (rentabilitate și risc) se apelează la teoria probabilităților care folosește o serie de indicatori statistici dintre care cei mai semnificativi sunt următorii:

1)Valoarea minimă, maximă și medie a rentabilității, unde valoarea medie este dată de relația:

$$m = \sum_{i=1}^n r_i p_i$$

unde r_i sunt rentabilitățile individuale, iar p_i sunt probabilitățile de apariție

2)Dispersia este dată de relația:

$$\sigma^2 = \sum_{i=1}^n (r_i - m)^2 p_i$$

Aceasta se poate asimila cu riscul ca rentabilitatea efectivă să se abată de la valoarea medie. Măsura cea mai utilizată a acestei dispersii este abaterea medie pătratică.

3)Abaterea medie pătratică este dată de relația:

$$\sigma = \sqrt{\sigma^2}$$

Aceasta măsoară gradul de reprezentativitate al mediei. Cu cât abaterea medie este mai mică, cu atât riscul titlului este mai redus deoarece variația rentabilității este mai redusă. Riscul unui titlu mobilier se poate calcula fie ca dispersia rentabilităților seriei dinamice față de rentabilitatea medie, fie ca abaterea medie pătratică a seriei respective, interpretabilă ca o creștere sau scădere a rentabilității, în funcție de valorile (pozitive, respectiv negative) abaterilor individuale.

4)Coeficientul de variație a lui Pearson este dat de relația :

$$v_x = \sigma_x / m_x$$

Acest coeficient măsoară întinderea distribuției probabilităților de apariție a rentabilităților unui titlu și reprezintă o modalitate de măsurare a riscului: cu cât coeficientul este mai mare cu atât distribuția este mai întinsă, abaterea medie pătratică este mai mare și riscul este mai important.

5)Covarianța este dată de relația:

$$\sigma_{ip} = m[(r_i - m(r_i))(r_p - m(r_p))] \text{ care în urma calculelor devine } \sigma_{ip} = \frac{1}{n} \sum (r_i - m(r_i))(r_p - m(r_p)) \text{ unde } r_i$$

este rentabilitatea individuală a unui titlu, r_p este rentabilitatea generală a indicelui pieței bursiere, iar n este numărul titlurilor individuale luate în calcul.

Relația dintre rentabilitate și risc

Riscul este dependent de realizarea posibilă a unor evenimente cu grad mare de incertitudine sau aleatorii, care definesc natura și evoluția piețelor financiare și, în particular, ale burselor de valori. Analiza riscului unui titlu mobilier se face în condiții de incertitudine, utilizând, în cele mai multe cazuri, calculul probabilistic. Măsurarea riscului cu ajutorul probabilităților constă în acordarea de grade de adevăr cuantificate (adică probabilități obiective sau subiective după caz) modificărilor posibile ale valorilor elementelor variabile sau aleatorii, care intervin într-un calcul global. Aceasta pleacă de la principiul potrivit căruia modificările determină stări diferite și deci măsurați ale consecințelor (de exemplu: asupra rentabilității unei investiții pe piața bursieră). Riscul aferent unei variabile financiare poate fi determinat dacă se cunoaște distribuția probabilităților acesteia.

Astfel, pentru a estima riscul ce însoțește profitul care se obține în urma unui plasament de titluri, trebuie studiate mai întâi contribuțiile tuturor factorilor care influențează obținerea profitului și, ca urmare, stabilirea unei distribuții a probabilităților corespunzătoare.

Să considerăm 2 cazuri în care se previzionează evoluția rentabilității astfel:

Cazul A

Rentabilitate	Probabilitate
7%	0,05
8%	0,10
9%	0,20
10%	0,30
11%	0,20
12%	0,10
13%	0,05

Cazul B

Rentabilitate	Probabilitate
9%	0,30
10%	0,40
11%	0,30

Rentabilitatea așteptată este media rentabilităților ponderate cu probabilitățile de apariție a lor: în cazul A 10% și în cazul B tot 10%. Se observă că avem aceeași medie în ambele cazuri, dar împrăștierea valorilor e mult mai mare în cazul A:

	Abaterea medie pătratică	Dispersia
Cazul A	1,45	2,10
Cazul B	0,77	0,60

Variabilitatea rentabilității în jurul valorii medii este o descriere cantitativă a riscului.

Ibbotson Associates a făcut un studiu asupra următoarelor 4 tipuri de portofolii de valori mobiliare:

- 1) un portofoliu de bonuri de tezaur (cu o maturitate sub un an)
- 2) un portofoliu de obligațiuni guvernamentale (pe termen lung)
- 3) un portofoliu de obligațiuni emise de diverse firme (pe termen lung)
- 4) un portofoliu de acțiuni obișnuite a 500 de firme

Aceste portofolii oferă grade diferite de risc. Pe perioada 1926-1988 s-au calculat rentabilitățile anuale medii, rentabilitățile reale medii (ce țin cont de rata inflației), abaterea medie pătratică și dispersia și s-au obținut următoarele rezultate:

Portofoliu	Rentabilitatea anuală medie	Rentabilitatea anuală reală medie	Abaterea medie pătratică	Dispersia
1	3,6	0,5	3,3	10,9

2	4,7	1,7	8,5	72,3
3	5,3	2,4	8,4	70,6
4	12,1	8,8	20,9	436,8

Se observă că rentabilitățile obținute de investitori variază în funcție de riscurile asumate. Piața financiară este riscantă datorită varietății rentabilităților pe care le oferă. Datele prezentate sunt o medie pentru o perioadă de peste 60 de ani, în prezent rentabilitatea la bonurile de tezaur, de exemplu, se situează în jur de 8%, cu mai mult de 4% față de rentabilitatea medie calculată de Ibbotson Associates.

Estimarea rentabilității cu ajutorul legii normale de probabilitate

Din studiile care s-au efectuat asupra rentabilităților valorilor mobiliare s-a ajuns la concluzia că legea normală constituie o bună aproximare a distribuției observate.

O variabilă aleatoare urmează legea normală de probabilitate dacă densitatea ei de probabilitate

este dată prin $f: \mathbb{R} \rightarrow [0, +\infty)$, $f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m)^2}{2\sigma^2}}$ ($m \in \mathbb{R}$, $\sigma > 0$).

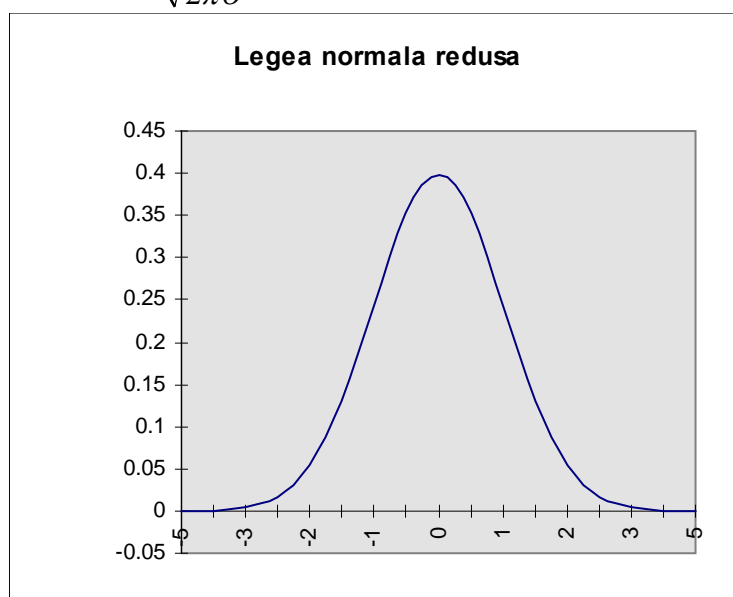


Fig. 5. Legea normală redusă de probabilitate

Legea normală se notează cu $N(m, \sigma)$ unde m este media, iar σ este abaterea medie pătratică.

O formă simplificată a acestei legi este Legea normală redusă $N(0, 1)$ în care media este 0 ($m=0$), iar dispersia 1 ($\sigma^2=1$) ilustrată în figura 5.

Cu ajutorul acestei legi putem determina care este probabilitatea ca o valoare mobilă:

1. să înregistreze o rentabilitate mai mare decât o rentabilitate dată r_1 , aplicând formula:

$$P(x \geq r_1) = \int_{r_1}^{\infty} f(x) dx$$

2. să se încadreze între două limite de rentabilitate (r_1 și r_2), aplicând formula:

$$P(r_1 \leq x \leq r_2) = \int_{r_1}^{r_2} f(x) dx$$

Fie z un multiplicator al abaterii medii pătratice aferent diferenței dintre rentabilitatea dorită și rentabilitatea medie: $(X-m)=z\sigma$. Folosind tabele matematice obținem probabilitățile ca rata rentabilității să se abată cu diferite valori de la rentabilitatea medie astfel:

$$P(X=m \pm 0,5\sigma) = 0,383; P(X=m \pm \sigma) = 0,683.$$

Deoarece $z=(X-m)/\sigma$ rezultă că:

$$P(X \geq A) = P((x-m)/\sigma \geq (A-m)/\sigma) = P(z \geq (A-m)/\sigma).$$

Pentru exemplificare se consideră o acțiune ce a avut un procent de variație zilnică urmând o lege normală de probabilitate cu $m=0,32\%$ și $\sigma=0,06\%$. Dorim să găsim probabilitatea ca procentul variației zilnice să fie între $0,3\%$ și $0,325\%$. Prin efectuarea calculelor, se obțin următoarele rezultate:

$$X=0,3 \rightarrow z=(0,3-0,32)/0,06=-0,33$$

$$X=0,325 \rightarrow z=(0,325-0,32)/0,06=0,08$$

$$P(z \leq -0,33) = P(z \geq 0,33) = 0,3707$$

$$P(z \geq 0,08) = 0,4681$$

$$P(0,3 \leq X \leq 0,325) = P(-0,33 \leq z \leq 0,08) = 1 - (0,3707 + 0,4681) = 0,1612$$

3.1.5. Concluzii privind analiza fundamentală

Bursa este, prin excelență, un spațiu al incertitudinii, al aproximării și mizei. Ca atare, valoarea de piață nu se poate determina cu certitudine prin simpla aplicare a unei formule; tot ceea ce putem obține prin analiza fundamentală e o imagine a valorii intrinseci a titlurilor. Pe baza celor prezentate mai sus, *autorul desprinde următoarele concluzii*:

- Orice estimare favorabilă privind activitatea unei firme duce la creșterea cursurilor acțiunilor, după cum orice veste negativă are un efect contrar.
- Valoarea de piață este determinată nu numai de raționalitatea economică ci și de o serie de factori subiectivi, extraeconomi, de atitudine subiectivă a indivizilor, de manevrele speculative.
- Valoarea intrinsecă a unui titlu constituie un indicator deosebit de important pentru investitorii la bursă, datorită posibilității de a elimina influența factorilor psihologici când se analizează fenomenul bursier, în vederea prognozării tendințelor pe piața de capital. Obiectivul calculării valorii intrinseci a unui titlu îl constituie posibilitatea determinării oscilațiilor prețului respectivului titlu la bursă, în sus sau în jos față de această valoare și, ca urmare, luarea unei decizii de vânzare sau cumpărare.
- Variațiile prețului, pe anumite perioade de timp, de la valoarea fundamentală au fost, de cele mai multe ori, interpretate ca reflexe ale modificărilor pe care le cunoaște estimarea valorii fundamentale, ca urmare a unui surplus de informații referitoare la starea economică a firmei cotate. Această teorie a fost însă infirmată de crahurile bursiere din 1929 și 1987.
- Importanța analizei fundamentale provine din faptul că analistul financiar va avea rezultate mai bune decât un investitor care acționează obișnuit, în măsura în care primul poate identifica mai rapid situațiile în care abaterea între cursul unui titlu și valoarea sa fundamentală este susceptibilă de a prezenta evenimente importante pe piață și a evalua influența acestora asupra valorilor mobiliare.
- Determinarea valorii intrinseci se realizează printr-o gamă variată de metode, de la cele mai simple, până la unele deosebit de complexe, bazate pe calcule laborioase. Prin urmare, se impune folosirea aplicațiilor informatice în analiza bursieră, acestea simplificând mult activitatea analiștilor.

3.2. ANALIZA TEHNICĂ

3.2.1. Caracteristici generale

Analiza tehnică (grafică, chartistă) vizează studiul graficelor de curs, evoluția trecută putând da informații asupra mișcărilor viitoare de curs.

Analiztii tehnici nu se interesează de comportamentul societății, de ce oferă pe piață, de bilanțuri sau dividende. Ei susțin că este posibil de dedus din aceste “curbe” ale trecutului evoluția viitoare a cursului, acestea reprezentând instrumente de previziune.

În conjuncție cu analiza regresională, tehnica analizei trendului este foarte folositoare. Frecvent, analiza trendului seriilor de timp utilizează analiza regresională. Ne referim la analiza regresională când studiem gradul de corespondență între două variabile și vorbim de analiza trendului când examinăm comportarea unui indicator în timp.

Analiztii tehnici încearcă să anticipeze viitorul pe baza seriilor de date pe care le oferă evoluțiile trecute ale acțiunilor. Evoluția prețului în trecut tinde să se repete și în viitor. Ei susțin că prețul depinde de cerere și de ofertă și are puțin în comun cu valoarea intrinsecă. Rezultatul este mișcarea cursului, tendințe pe diferite perioade de timp. Schimbările de trend reprezintă după opinia lor schimbări în balanța cerere-ofertă, uneori reflectând doar forțe emoționale.

Analiza tehnică este folosită mai mult decât un supliment la analiza fundamentală, ea servește la hotărârea momentului vânzării sau a cumpărării. Analistul trebuie:

- 1) să identifice trendul
- 2) să cunoască momentul când un trend ajunge în final și prețul o ia în direcție opusă.

Figurile chartiste cele mai semnificative sunt: Curba continuă, Tendințe și tunele, Cap și umeri, Formația M, Formația W, etc.

3.2.2. Tendințe și tunele; Cap și umeri; Formațiile M și W

În general, analiza cursurilor bursiere duce la distingerea unor tendințe ce pot fi de creștere, de scădere sau aproximativ constante. În funcție de orizontul de timp considerat putem avea o tendință primară (urmărită pe perioade de peste un an) și tendințe secundare (reflectă evoluții de câteva luni). În cadrul unei tendințe primare crescătoare, putem distinge, pe perioade mai scurte, tendințe secundare descrescătoare așa cum rezultă din următoarea reprezentare grafică (figura nr. 6):

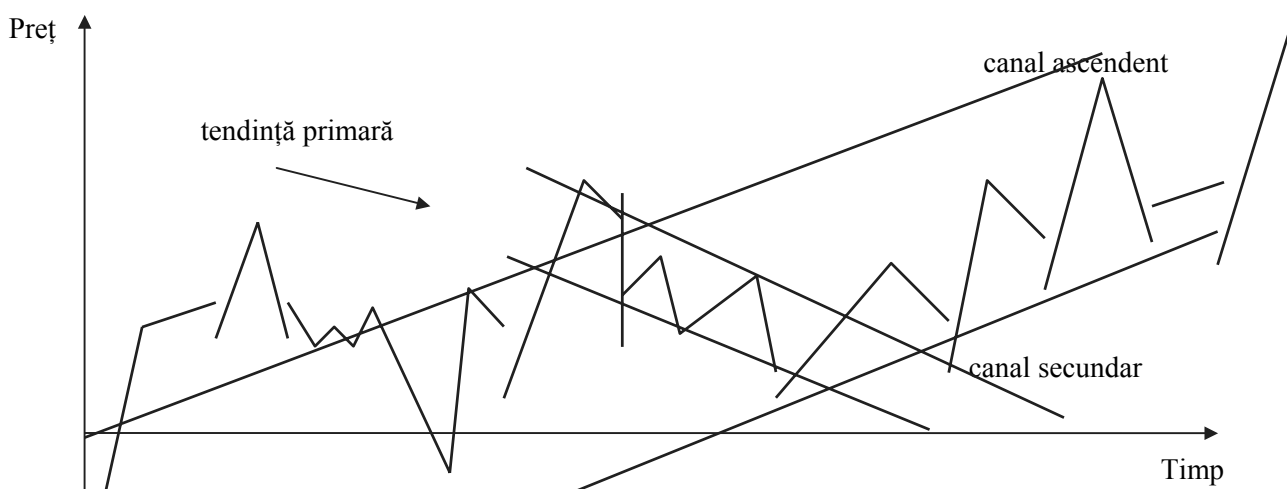


Fig. 6 Tendințe și tunele

În figura 7 sunt ilustrate formațiile cap și umeri, care pot să aibă următoarele configurații:

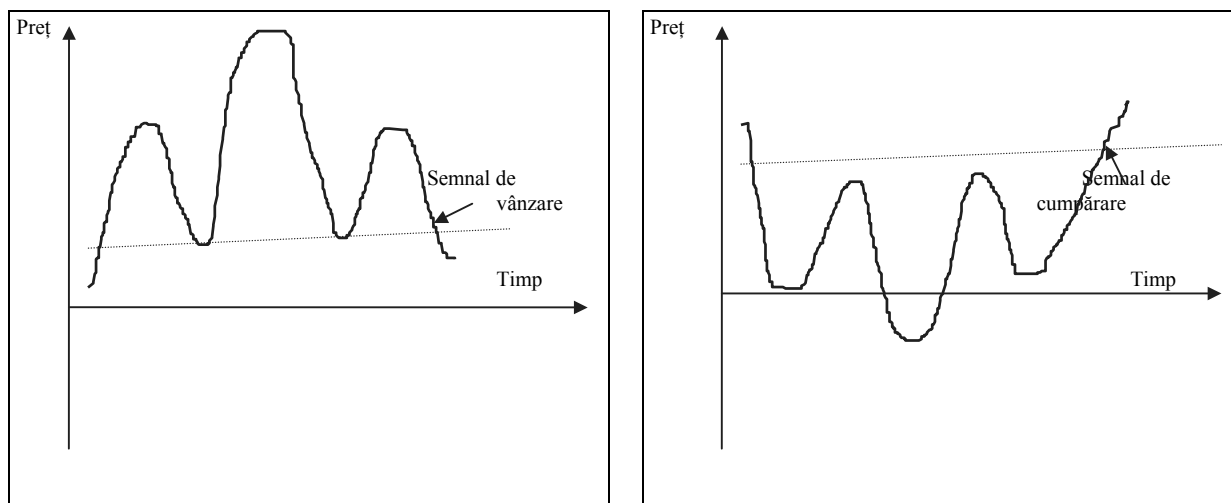


Fig. 7. Cap și umeri în cele două configurații

Aceste formații sunt caracteristice în special pieței americane și britanice. Prima configurație prezintă următoarele caracteristici³⁰:

- umărul stâng marchează sfârșitul unei perioade intensive de cumpărare de titluri, care, după o scurtă perioadă de acalmie, înregistrează un nou puseu de creștere a cursurilor, înainte de scăderea bruscă a acestora. După scăderea cursurilor, care include umărul stâng, se înregistrează o perioadă de cumpărări, care vor determina creșterea cursurilor către un nou vârf (capul), superior ca valoare umărului stâng. Urmează o nouă scădere până la un prag minim, a cărui valoare poate fi sub valoarea umărului stâng;
- umărul drept reprezintă o nouă repriză de accelerare a cumpărării de titluri, dar al cărei maxim nu va atinge nivelul înregistrat de capul formației. Acesta reprezintă un semnal de saturare a pieței și este urmat, în general, de o scădere continuă a prețului, fiind în același timp un semnal de vânzare;
- linia gâtului se obține unind minimele din formație și reprezintă nivelul de sprijin, sub care scăderea cursurilor indică un fenomen pe termen lung.

Semnalul de vânzare e dat atunci când cursul străpunge “linia gâtului” cu mai mult de 3%. În general se previzionează o scădere egală cu creșterea de la “linia gâtului” până la punctul cel mai înalt al capului. În cazul în care se anticipează existența acestei configurații se poate declanșa mai repede semnalul de vânzare (în momentul de maxim al celui de al treilea val).

Această figură poate avea și o configurație inversă, caz în care străpungerea “liniei gâtului” constituie într-un semnal de cumpărare. Creșterea din acest moment poate fi previzionată ca egală cu distanța dintre “linia gâtului” până la punctul cel mai scăzut al capului.

Durata unei asemenea formații este de 2-3 luni.

Formațiile *M* și *W* sunt configurații simetrice și se pot reprezenta grafic astfel (figurile 8 și 9):

³⁰ [Ghilic97] Ghilic-Micu B. -Bursa de Valori, Editura Economică, 1997

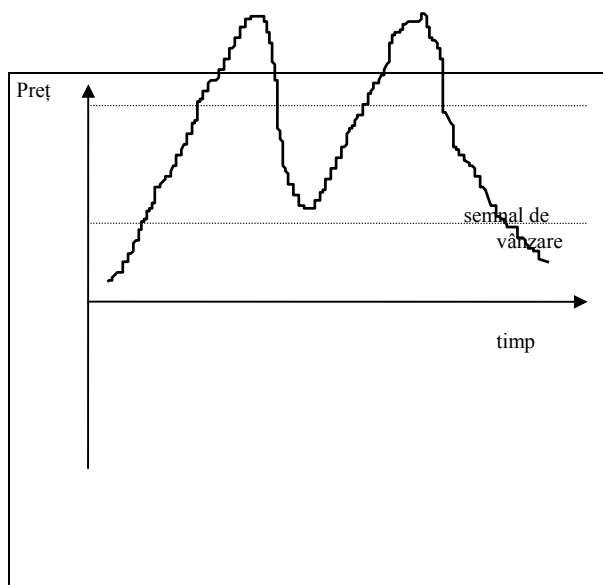


Fig. 8. Formația M

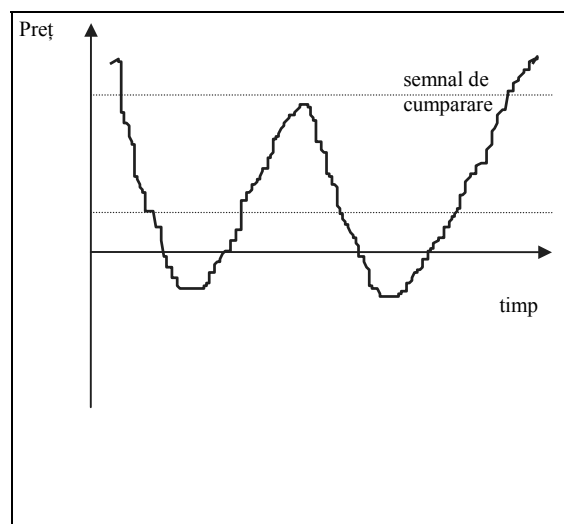


Fig. 9. Formația W

Interpretarea graficelor conduce la concluzia ca titlul mobilier a încercat sa depășească (să scadă sub) un anumit nivel maxim (respectiv minim), dar nu a reușit.

Este probabil ca la a doua încercare, investitorii care au fost induși în eroare de prima reacție și au cumpărat titluri se vor grăbi să le înstrăineze, determinând astfel o scădere. Câteodată, investitorii trebuie să se aștepte la a treia încercare de creștere a cursurilor, care va genera reacția cumpărătorilor: este cazul graficului cu 3 vârfuri superioare. Scăderea însă este previzibilă încă de la debutul ei. Durata care separă două vârfuri succesive este de 2 săptămâni (maxim o lună). Graficele cu 2 sau 3 vârfuri inferioare sunt mișcări simetrice ale tendinței cursurilor bursiere indicând o reacție inversă. Ele sunt adesea întâlnite la titlurile de pe piața franceză.

3.2.3. Teoria Dow

La sfârșitul secolului trecut, Charles Dow a formulat o ipoteză conform căreia piața financiară nu urmează o tendință aleatoare, ci este influențată de 3 trenduri ciclice care îi ghidează direcția. Urmărind aceste trenduri, direcția generală a pieței poate fi previzionată.

Dow clasifică aceste cicluri în: primare, secundare și minore (figura nr. 10). Trendul primar direcționează întreaga piață, pe termen lung, într-o direcție ascendentă (*bull market*) sau descendentă (*bear market*). Trendul secundar acționează ca o restrângere a forței trendului primar, ținând de regulă de la câteva săptămâni până la câteva luni. Trendurile minore sunt fluctuații zilnice ale pieței. Acestea au valoare analitică mică din cauza duratei și a variației în amplitudine mici.

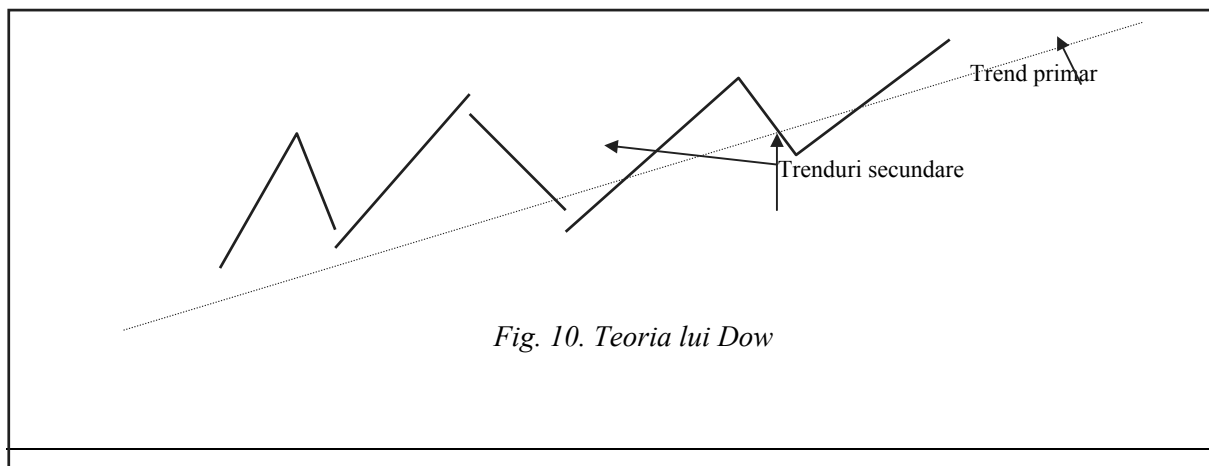


Fig. 10. Teoria lui Dow

Trend-ul primar sau principal este ciclul pe termen lung care conduce piața în sus sau în jos, putând acoperi o perioadă de până la patru ani. Acest tip de fluctuație indică tendința generală a pieței.

Tendințele secundare se manifestă sub forma unor forțe restrictive asupra trend-ului primar, tinzând să corecteze deviațiile de la limitele generale. Cu alte cuvinte, acestea acționează în contrasensul tendinței pe termen lung, adică sunt mișcări ascendente până când se atinge o fază pe termen lung descendentă, și invers. Durata acestor trend-uri secundare se poate întinde de la cel puțin două săptămâni la câteva luni.

Trend-urile minore sunt de fapt fluctuațiile zilnice. Sunt considerate neglijabile și cu o valoare analitică foarte scăzută datorită duratei scurte și variației reduse în amplitudine.

Dow afirmă că o piață este în creștere atunci când creșterile succesive sunt atinse după corecții secundare și când creșterile secundare avansează dincolo de scăderile anterioare (deci sunt încadrate într-un trend ascendent). Teoria mai cere ca scăderile corective să fie cu o durată și amplitudine mai scurte decât creșterile corective. Situația inversă va fi valabilă pentru o piață în scădere ("bear market").

Această teorie are două neajunsuri: pe de o parte, fluctuațiile secundare nu sunt foarte clare iar pe de altă parte, metoda nu face decât să confirme o schimbare în evoluția pe termen lung a unei piețe bursiere (și chiar cu o anumită întârziere), dar în nici un caz să o anticipeze.

3.2.4. Analiza cu ajutorul mediilor mobile

Utilizarea mediilor mobile ca recurs la teoria lui Dow are ca scop evidențierea schimburilor de tendință ale pieței. Pe baza seriei de date ce reprezintă cursurile bursiere, se construiește o serie de medii mobile după formula:

$$M_t = \frac{1}{N} \sum_{i=t-N+1}^t C_i$$

unde M_t este media mobilă în ziua t ; N este perioada de timp (zile); C_i este cursul bursier în ziua i .

Durata N a mediei mobile se alege suficient de mare (în general 200 zile, perioada de glisare cel mai frecvent utilizată) pentru a putea desprinde evoluția pe termen lung. Peste graficul reprezentând evoluția cursului bursier se suprapune curba reprezentând evoluția mediei mobile (figura nr.11).

Interpretarea rezultatelor conduce la stabilirea următoarelor tipuri de semnale ce indică schimbarea de tendință a pieței³¹:

- atunci când curba mediilor mobile este intersectată de curba observațiilor bursiere
- atunci când curba mediilor mobile își schimbă alura
- atunci când curba cursurilor bursiere prezintă variații intermediare sporadice până la nivelul semnalului precedent

Utilizate pentru un titlu anume, mediile mobile servesc nu numai pentru determinarea tendinței de evoluție pe termen lung a variațiilor de curs bursier, ci constituie și un instrument de previziune ce poate indica momentul oportun pentru vânzarea sau cumpărarea unui titlu.

³¹. [Ghilic97] Ghilic-Micu B. -Bursa de Valori, Editura Economică, 1997

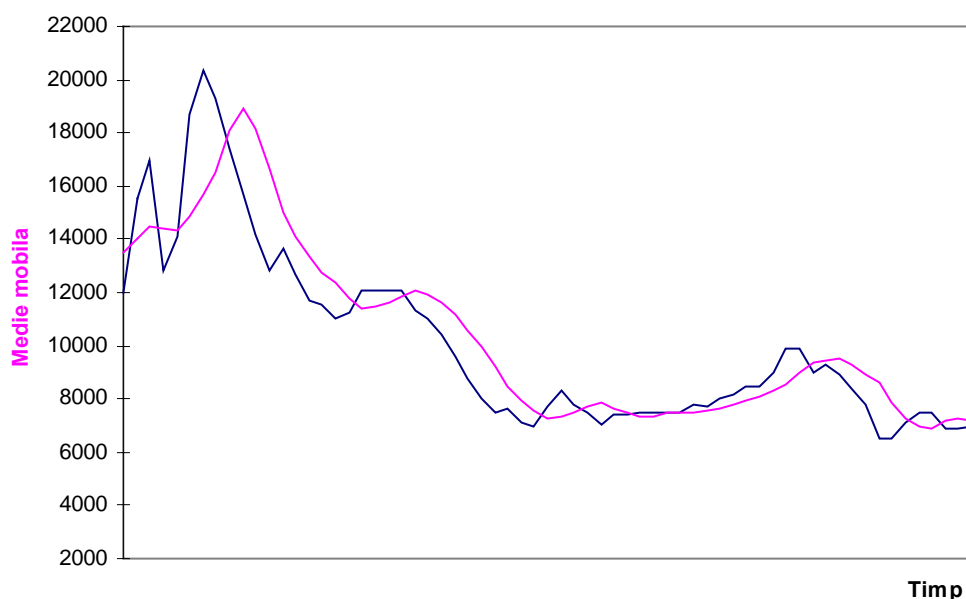


Fig. 11. Analiza cu ajutorul mediei mobile

Presupunând un titlu într-o fază ascendentă, ajustarea cu ajutorul mediilor mobile presupune intersectarea acesteia cu evoluția cursului. Intersecția va fi cu atât mai îndepărtată cu cât intervalul mediei mobile este mai mare. Este posibilă o modificare a tendinței atunci când curba mediilor mobile este încă crescătoare, iar curba cursurilor bursiere este deja descendentă. În momentul intersecției curbei mediilor mobile cu curba cursului bursier, modificarea de tendință nu este semnificativă decât în măsura în care curba mediilor mobile se aplatizează. Acest lucru este un semnal care indică sfârșitul fazei de creștere și furnizează o indicație de vânzare.

Dacă graficul cursurilor bursiere prezintă o pantă pozitivă și superioară celei a mediilor mobile, atunci există o cerere de titluri nesatisfăcută, care conduce la creșterea cursului respectivei valori mobiliare. În cazul contrar, dacă panta cursurilor este negativă și inferioară mediilor mobile, piața prezintă o ofertă excedentară dintr-un titlu, care conduce la scăderea prețului acestuia.

Pentru previziunea cursului bursier, analiștii folosesc un sistem de reguli aplicabile la mediile mobile de 200 zile (perioadă de glisare cel mai frecvent utilizată). Acestea sunt următoarele³²:

- Se consideră că există un semnal de cumpărare când:
 - curba mediilor mobile se redresează sau crește după o scădere și este traversată de curba cursurilor de jos în sus;
 - curba cursurilor intersectează curba mediilor mobile, care cunoaște o creștere continuă;
 - curba cursurilor, situată deasupra mediilor mobile, înregistrează o creștere în concordanță cu aceasta;
 - curba cursurilor scade brusc sub cea a mediilor mobile, care prezintă o pantă descendentă; în această situație, se poate aștepta o revenire a cursurilor către valoarea mediei mobile.
- Un semnal de vânzare apare atunci când:
 - curba mediilor își modifică sensul sau scade după o creștere, iar curba cursurilor o intersectează în punctul maxim;
 - curba cursurilor devansează curba mediilor mobile, chiar când prezintă o formă descrescătoare;
 - curba cursurilor tinde, în creștere, către curba mediilor mobile și înregistrează apoi o scădere;
 - curba cursurilor urcă rapid deasupra curbei crescătoare a mediilor mobile; este de prevăzut o reacție de scădere pe termen scurt.

3.2.5. Indicatorii urmăriți de analiștii tehnici

³² [Ghilic97] Ghilic-Micu B. -Bursa de Valori, Editura Economică, 1997

Cele două variabile referitoare la grupe de acțiuni sau acțiuni individuale urmărite de tehnicieni cu cel mai mare interes sunt evoluția prețului și volumului tranzacțiilor. În examinarea influenței pieței asupra prețului acțiunilor în general tehnicienii urmăresc anumite semnale precum: noi valori maxime și minime, creșteri ale unor acțiuni însoțite de scăderi ale altora, prețurile celor mai “active” acțiuni etc. Urmărirea doar a indicelui bursier poate fi adesea derutantă (se întâmplă ca indicele să prezinte o creștere ușoară și cu toate acestea un număr mare de acțiuni să fie în scădere).

Măsurarea creșterilor și a scăderilor trebuie să urmărească spre ce se îndreaptă marea parte a acțiunilor. Unul din indicatorii urmăriți este egal cu diferența dintre numărul zilnic al societăților care înregistrează creșteri și numărul celor care sunt în scădere. Acest număr se urmărește cumulat și se compară cu evoluția indicelui bursier. O piață în creștere este acompaniată de un număr în creștere de acțiuni ce înregistrează noi valori maxime și de un număr scăzut de acțiuni ce înregistrează valori minime. Volumul tranzacțiilor este folosit de cei mai mulți tehnicieni pentru a previziona orice schimbare în preț. Acest volum reprezintă o funcție a cererii și ofertei care poate da semnale privind punctele de inversiune ale pieței sau ale acțiunilor individuale. În general volumul scade înaintea momentului unor dezechilibre ale prețului și crește puternic în momentul în care piața începe să-și revină.

Conform lui Charles C. Ying³³, volumul tranzacțiilor prezintă următoarele caracteristici:

- un volum mic (mare) e de obicei acompaniat de o scădere (creștere) a prețului;
- dacă volumul a scăzut (respectiv crescut) consecutiv pentru o perioadă de cinci zile de tranzacționare va exista o tendință ca prețul să scadă (respectiv să crească) în următoarele patru zile de tranzacționare.

Totuși, previziunea prețului cere examinarea atât a trendului prețului cât și a fluctuațiilor în volumul tranzacțiilor.

3.2.6. Concluzii privind analiza tehnică

Economiștii, analiștii financiari, precum și investitorii sunt în permanență interesați în a previziona tendințele (de creștere sau de scădere) pe piețele de capital.

Analiza tehnică este procesul de analiză a istoriei cursului valorilor mobiliare, într-un efort de a determina cursurile viitoare probabile. Un chartist demodat ar putea defini analiza tehnică drept un proces în care istoria se repetă, pe când alții l-ar putea defini ca un proces prin care învățăm din trecut.

În opinia autorului, prin analiza tehnică se încearcă stabilirea momentelor în care piața bursieră trece din starea de creștere în cea de scădere sau invers. Analiza tehnică a pieței de valori mobiliare a fost dezvoltată prin excelență de practicieni, fiind din ce în ce mai utilizată. Ea se bazează exclusiv pe studiul datelor interne ale pieței bursiere. Fundamentarea sa constă în faptul că toți factorii economici, financiari, politici și psihologici care influențează cursurile bursiere sunt conținuți în fluxurile cererii și ofertei și că observarea volumului tranzacțiilor, precum și a variațiilor de cursuri sunt suficiente pentru a prevedea evoluția acestora. Utilizarea acestei metode este însă incompatibilă cu ideea unei evoluții aleatoare a cursurilor, cu conceptul de “piață la întâmplare”.

Așadar, constatăm că analiza tehnică poate fi folosită mai mult decât un supliment al analizei fundamentale. Analiza tehnică devine importantă în indicarea momentului lansării ordinului de vânzare sau cumpărare, ordin fundamentat, de altfel, pe previziunile referitoare la risc și rentabilitate. Utilizarea indicatorilor analizei tehnice are un rol foarte important pentru un operator de piață, deoarece îi pot oferi o imagine reală și fidelă asupra pieței, îi pot semnaliza cele mai oportune momente de a vinde și de a cumpăra, îi pot indica trendul urmat de piață, ajutându-l astfel să pună bazele unor decizii reușite de tranzacționare și de intrare sau ieșire din piață și să obțină profituri prin lichidarea poziției la momentul oportun.

³³ [Ying96] Ying C.C. -Stock Market Prices and Volumes of Sales, *Econometrica*, July, 1996.

3.3. TEORIA PIETELOR EFICIENTE (EFFICIENT MARKET HYPOTHESIS -EMH)

3.3.1. Generalități

Teoria piețelor eficiente este o teorie statistico-matematică de studiere a comportamentului cursurilor bursiere și deci a structurii pieței de valori mobiliare. Reprezentanții teoriei pieței eficiente susțin că nu se poate prevedea cursul viitor pe baza evoluției trecute. Această teorie afirmă că, în condițiile pieței moderne, dominată de calculatoare, baze mari de date, de mijloace de comunicare rapidă și de analiști competenți și numeroși, valoarea intrinsecă a unei acțiuni nu este altceva decât cursul său de piață, deci o acțiune valorează atâta cât costă. Piața asimilează instantaneu toate informațiile utile, de orice natură, și le reflectă tot instantaneu în cursul momentan. Deci prețurile titlurilor reprezintă cea mai bună estimare posibilă a valorii intrinseci a titlurilor, ținând seama de toată informația public disponibilă, iar orice schimbare de preț este independentă de evoluția trecută.

Există trei forme ale teoriei pieței eficiente, distincția făcându-se în funcție de modul în care este înțeleasă informația disponibilă. Eficiența nu poate fi testată empiric decât în raport de trei categorii de informații, sub forma a trei teste și corespunzând unor niveluri sau grade diferite de eficiență informațională astfel:

1) testele de eficiență slabă reprezintă prima categorie de informații și cuprind doar seria cronologică a cursurilor bursiere. Forma slabă a EMH (*Efficient Market Hypothesis*) susține că prețurile titlurilor reflectă în întregime toate informațiile date de piață privind evenimentele trecute. Aceasta conduce spre ideea conform căreia cursul acțiunilor nu conține informații care să poată fi folosite pentru a obține profituri suplimentare față de cele obținute printr-o alegere aleatoare. Deci studiul și utilizarea cursurilor din trecut nu pot fi utilizate pentru a previziona mișcările viitoare de curs ceea ce conduce spre inutilitatea analizei tehnice.

Studii statistice au relevat faptul că modificările cursului titlurilor de-a lungul timpului sunt analoage unor mișcări întâmplătoare (*Random Walk*).

2) testele de eficiență medie (semitare) reprezintă a doua categorie de informații și cuprind informațiile public disponibile. Forma medie a teoriei EMH o include pe cea slabă.

Cât de aproape e o piață de forma medie de eficiență depinde în parte de cât de repede sunt asimilate informațiile. Aceasta depinde de interesul investitorilor, care diferă de la o acțiune la alta; există o competiție intensă în cazul marilor companii ceea ce conduce spre o eficiență mai ridicată decât în cazul pieței acțiunilor companiilor mici, care atrag mai puțin atenția.

Un număr de studii au arătat că piața nu reacționează imediat, în ziua anunțării informațiilor favorabile sau nefavorabile referitoare la o anumită societate. Există un decalaj între momentul anunțării și momentul în care informația e asimilată și reflectată în cursul acțiunii.

3) testele formei tari de eficiență urmăresc dacă prețurile titlurilor se ajustează la toate informațiile posibile, publice sau nu și dacă viteza de ajustare e destul de mare pentru a elimina posibilitatea unor câștiguri mărite. Forma tare include forma medie deoarece informația public disponibilă este inclusă în totalitatea informațiilor posibile. Forma tare de eficiență duce la concluzia că este imposibil să se obțină randamente superioare decât într-un mod cu totul aleator.

Ipoteza pe care se bazează forma tare de eficiență e una puțin plauzibilă datorită faptului că unii investitori (*insiders*) au acces la informații privilegiate; ei pot obține astfel profituri ridicate folosind aceste informații. E greu să se elimine tranzacțiile ce se bazează pe informații confidențiale. Din acest punct de vedere, piața este inefficientă. Studiile au confirmat faptul că piețele nu respectă condițiile eficienței în forma tare.

În concluzie, o piață eficientă este una în care un număr mare de vânzători și cumpărători reacționează prin intermediul unui sistem sensibil și eficace, determinând stabilirea unor prețuri care

reflectă pe deplin și practic instantaneu, tot ceea ce se poate cunoaște despre perspectivele companiilor ale căror titluri sunt tranzacționate³⁴.

Teoria pieței eficiente are la bază și un paradox: pentru ca ea să fie adevărată, investitorii trebuie să facă eforturi pentru a analiza rapid toate informațiile disponibile și de a le reflecta în preț -acest lucru conduce spre ideea că investitorii nu pot obține profituri superioare datorită faptului că munca analiștilor de determinare a valorii intrinseci a unei acțiuni este inutilă întrucât cursul o reflectă deja, ceea ce conduce spre lipsa interesului investitorilor care se va reflecta printr-o ineficiență a pieței.

William Sharpe de la Universitatea din Stanford, unul din cei mai mari adepți ai acestei teorii în anii 1960 și 1970 a recunoscut ulterior că piața nu este perfect eficace și în orice caz, nu este instantanee.

3.3.2. Teoria “pieței la întâmplare” (“random walk”)

Această teorie încearcă să stabilească dacă evoluția în timp a cursurilor bursiere, studiate prin procedee matematice și statistice, are un *caracter aleator* sau nu. Concluzia este că această evoluție este într-adevăr aleatoare, în sensul probabilistic al termenului. Aceasta înseamnă că este imposibil de dedus cursul viitor pe baza cursurilor trecute și că pretențiile previzioniste ale analiștilor sunt fără fundament științific. Printre protagoniștii cei mai cunoscuți ai acestei teorii se găsesc Paul Cootner (de la M.I.T.), Burton Malkiel (Universitatea din Princeton), Eugene Fama (Universitatea din Chicago) și chiar Paul Samuelson, laureat al premiului Nobel pentru economie.

Teoria pieței la întâmplare confirmă faptul că schimbările succesive de preț sunt independente reciproc dar nu spune nimic despre trendurile pe termen lung sau cum sunt determinate nivelurile prețurilor.

Această teorie, cu toate numele de prestigiu a celor care o susțin, nu a reușit să taie aripile analiștilor. Într-adevăr, se observă deseori anumite tendințe ce se mențin un anumit timp și care contrazic această teorie, deci există momente când analiștii, pe baza studiilor și a graficelor, pot afirma cu o probabilitate ridicată evoluția cotațiilor viitoare ale unei anumite societăți.

Folosind generatorul de numere aleatoare al calculatorului, cursul unei acțiuni ce pornește de la valoarea de 100 u.m. și poate înregistra fie o creștere de 1 u.m. (cu o probabilitate de 0,5) fie o scădere de 1 u.m. (cu aceeași probabilitate) poate evolua conform figurii 12:

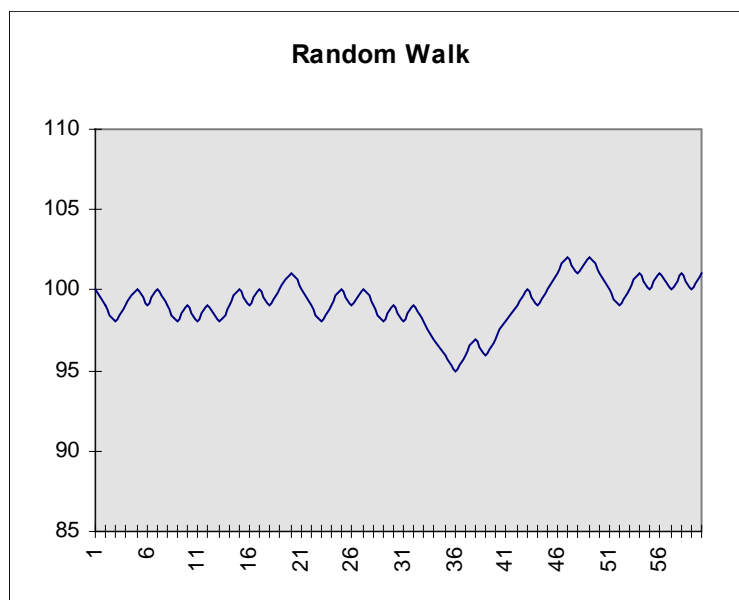


Fig. 12. Evoluția cursului unei acțiuni conform teoriei Random Walk

³⁴ [Popa93] Popa I. -Bursa, vol I, Colecția Bursa, București, 1993.

Asemenea simulări se pot face și după alte reguli: de exemplu se pot înregistra creșteri de 3% sau scăderi de 2,5% (fiecare cu o probabilitate de 0,5). Acest proces are o abatere pozitivă de 0,25% ($0,5 \times 3\% + 0,5 \times (-2.5\%) = 0,25\%$).

3.3.3. Concluzii privind teoria piețelor eficiente

Din studiul teoriei piețelor eficiente, *autorul desprinde următoarele concluzii:*

- Într-o piață eficientă, concurența antrenează situații în care, la fiecare moment, cursurile titlurilor mobiliare reflectă instantaneu și corect orice informație disponibilă. În orice moment, cursul unui titlu va fi o estimare suficient de bună față de valoarea sa reală.
- În viața economică reală, piețele au grade diferite de eficiență; în cazul piețelor dezvoltate, ineficiențele majore sunt rare. Ineficiența e reflectată prin diferențele dintre prețurile curente și valoarea intrinsecă a titlurilor; astfel de situații fiind rapid sesizate de participanții la piață. Dacă prețul e sub valoarea sa va crește cererea ceea ce va duce la creșterea prețului; dacă prețul e peste valoarea sa, va crește oferta, ducând la scăderea prețului. Într-o piață eficientă asemenea diferențe nu apar, în orice moment prețul trebuie să reflecte aproximativ valoarea intrinsecă.

Apreciem că teoria pieței aleatoare și a piețelor eficiente intră în competiție, în același timp, cu analiza fundamentală și cu analiza grafică. Analiza fundamentală nu este cu adevărat utilă decât în măsura în care utilizează informații care nu au influențat deja formarea cursurilor.

3.4. ANALIZA CURSULUI ACȚIUNILOR ȘI A OBLIGAȚIUNILOR

3.4.1. Variabilitatea rentabilității acțiunilor individuale

Fluctuațiile de curs ale valorilor mobiliare sunt influențate în general de modificările indicelui general al bursei de valori și în particular de anumite condiții specifice ale titlurilor. Surprinderea acestei caracteristici principale a valorilor mobiliare se face prin utilizarea modelului de piață, elaborat de H.Markovitz și dezvoltat ulterior de W.Sharpe. Ideea centrală a acestui model este aceea că fluctuațiile de curs ale valorilor mobiliare sunt influențate, în general, de modificările indicelui general al bursei și, în particular, de modificările condițiilor specifice ale societăților emitente ale titlurilor. Variabilitatea totală a rentabilității unui titlu are două componente³⁵:

-riscul sistematic (risc nediferențiat și nediversificabil) este determinat de influența pieței bursiere. El se referă la partea din variabilitatea totală cauzată de factori ce afectează prețul tuturor titlurilor. Schimbările economice, politice, sociologice sunt surse ale riscului sistematic. De exemplu, când se prevede o perioadă de recesiune, prețurile acțiunilor scad.

-riscul specific (risc diferențiat, individual, nesistematic și diversificabil) este determinat de influența caracteristicilor specifice ale fiecărui titlu în parte. Riscul specific este partea din riscul total care este unic pentru o firmă sau o industrie. Factori precum aptitudinile manageriale, comportamentul consumatorilor sunt cauze ale riscului specific. Riscul specific are și el două părți: riscul specific fiecărei acțiuni datorită evoluției economice a respectivei societăți (risc specific intrinsec) și riscul specific ramurii din care face parte emitentul (risc specific sectorial).

Astfel, în unele cazuri, putem asista la creșterea cursurilor tuturor titlurilor de pe piață (o dată cu creșterea indicelui bursier); în alte cazuri putem asista la scăderea acțiunilor tuturor titlurilor ce provin dintr-o anumită ramură, care, din diferite motive, se află în declin. În general, pe o piață eficientă, creșterile sau scăderile de curs au cauze ce țin de evoluția economică internă a emitentului.

3.4.2. Volatilitatea unei acțiuni

Între rentabilitatea individuală a unei acțiuni și rentabilitatea generală a pieței există o anumită corelație. Pentru a putea pune în evidență această corelație se reprezintă grafic punctele ce indică aceste rentabilități prin dreapta de regresie. Panta acestei drepte reprezintă *volatilitatea* respectivei acțiuni, respectiv sensibilitatea rentabilității ei la modificările rentabilității generale ale pieței. Cu cât punctele sunt mai adunate în jurul drepte de regresie cu atât corelația este mai puternică și implicit riscul sistematic este mai ridicat. Dacă punctele sunt împrăștiate față de dreapta de regresie cu atât corelația este mai scăzută și deci riscul sistematic are o pondere mai scăzută.

Fie $r_i = \alpha + \beta r_p + \varepsilon$ ecuația drepte de regresie, unde r_i este rentabilitatea individuală a acțiunii i , r_p este rentabilitatea pe piață, măsurată prin indicele general de bursă, β este coeficientul de regresie, de volatilitate numit și coeficient beta, iar ε este o variabilă aleatoare reziduală (variabilă de perturbație) specifică titlului respectiv deci prin care se măsoară riscul individual.

α și β se află prin metoda celor mai mici pătrate a lui Gauss. Rezolvarea sistemului ne conduce la următoarele valori:

$$\alpha = (\sum r_i \sum r_p^2 - \sum r_p \sum r_i r_p) / (n \sum r_p^2 - (\sum r_p)^2) \text{ și } \beta = (n \sum r_i r_p - \sum r_i \sum r_p) / (n \sum r_p^2 - (\sum r_p)^2) \text{ unde}$$

α are o importanță mai redusă (semnifică intersecția drepte de regresie cu axa ordonatelor și reprezintă rentabilitatea acțiunii i atunci când rentabilitatea generală de piață este nulă).

β are o importanță deosebită. În funcție de valoarea lui beta acțiunile se clasifică în trei grupe³⁶:

³⁵ [Anghelache92] Anghelache G., Dardac N., Stancu I. -Piețe de capital și burse de valori, Soc. Adevărul S.A., București, 1992.

³⁶ [Anghelache92] Anghelache G., Dardac N., Stancu I. -Piețe de capital și burse de valori, Soc. Adevărul S.A., București, 1992.

1) *acțiuni foarte volatile* (cu $\beta > 1$). O creștere (scădere) cu 1% a indicelui general al pieței bursiere antrenează o creștere (scădere) mai mare decât 1% a respectivelor acțiuni. În general industria electrică și electronică, a automobilelor, industria aparatelor casnice, industria chimică etc, au o sensibilitate mai ridicată, deci acțiunile lor fac parte din această categorie.

2) *acțiuni puțin volatile* (cu $\beta < 1$). O creștere (scădere) cu 1% a indicelui general al pieței bursiere antrenează o creștere (scădere) mai mică decât 1% a respectivelor acțiuni. Din această categorie fac parte acțiunile emise de întreprinderi din industria alimentară, industria medicamentelor, societățile de asigurări, întreprinderile producătoare de material rulant.

3) *acțiuni ce evoluează în același ritm cu piața* (cu $\beta = 1$). Se întâlnesc în întreprinderi de textile-încălțăminte, construcții industriale, bănci etc.

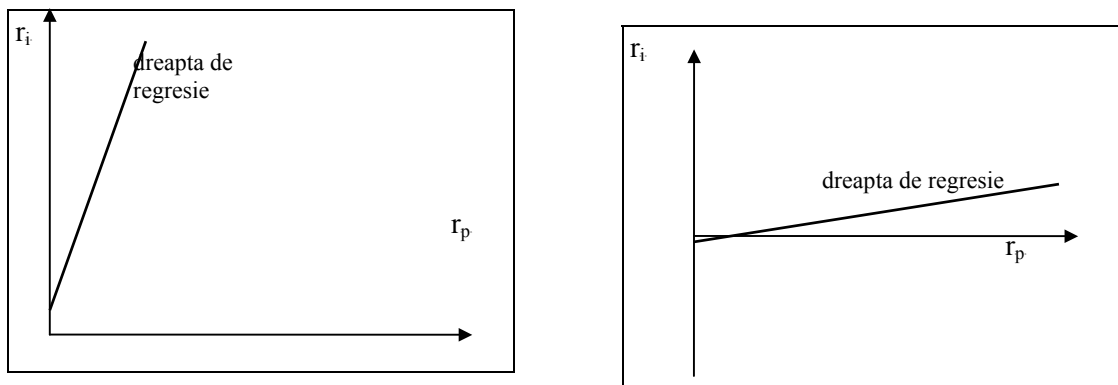


Fig. 13 Dreapta de regresie în cazul acțiunilor foarte volatile și respectiv puțin volatile

În figura 13 se observă că în cazul acțiunilor foarte volatile, dreapta de regresie prezintă un unghi față de ordonată mai mare de 45° , iar în cazul acțiunilor mai puțin volatile unghiul este de sub 45° .

Continuând calculele pentru β obținem că acest coeficient este egal cu covarianța σ_{ip} a ratelor rentabilității acțiunii i cu cea generală a pieței bursiere raportată la dispersia ratelor rentabilității generale:

$$\beta = \sigma_{ip} / \sigma_p^2.$$

3.4.3. Riscul unei acțiuni

Pentru a calcula riscul specific pentru o acțiune, trebuie să pornim de la matricea de variație și covariație care arată astfel:

$$M^2 = \begin{pmatrix} m_{00} & m_{01} \\ m_{10} & m_{11} \end{pmatrix}$$

unde $m_{00} = m(r_i^2) - m^2(r_i)$; $m_{01} = m_{10} = m(r_i r_p) - m(r_i) m(r_p)$; $m_{11} = m(r_p^2) - m^2(r_p)$.

Riscul specific unei acțiuni se determină folosind *coeficientul de corelație* C care măsoară gradul de dependență al acțiunii i față de piață utilizând formula:

$$C = \sqrt{1 - \frac{|M^2|}{m_{00} |M_{00}^2|}} = \sqrt{1 - \frac{m_{00} m_{11} - m_{01}^2}{m_{00} m_{11}}} = \frac{m_{01}}{\sqrt{m_{00} m_{11}}}$$

Acest coeficient ia valori între -1 și 1; pentru valori pozitive avem o dependență direct proporțională între acțiunea i și piața bursieră, iar pentru valori negative relația de dependență este invers proporțională. Pentru $C=0$ nu există nici o legătură între rentabilitățile individuale și cea de piață. Pentru $C=1$, variația rentabilității acțiunii i este integral și exclusiv datorată variației rentabilității generale a pieței bursiere (cea mai mare volatilitate și risc specific=0). Cu cât acest coeficient este mai apropiat de 1 cu atât dependența rentabilității acțiunii i față de rentabilitatea pieței este mai ridicată. Dacă $C=0,4$ atunci 40% din riscul ce însoțește această acțiune e determinat de dependența directă față de piață (volatilitatea ei) și 60% din risc e determinat de riscul ei specific (caracteristicile proprii).

3.4.4. Valoarea actualizată a unei obligațiuni

Valoarea actuală este echivalentul de azi al unor sume ce vor fi primite în viitor, adică ceea ce trebuie să plătim azi pentru a obține fluxurile viitoare. Actualizarea, ca procedeu de aducere la momentul 0 a unor valori virtuale în perioada $[0, n]$, este complementul metodologic al capitalizării, ca proces de sporire a capitalului în timp prin înglobarea continuă a produsului fructificării sale (profitul). În cazul obligațiunilor trebuie actualizate sumele reprezentând cupoanele de încasat și suma de rambursat (sau prețul de piață în cazul în care investitorul dorește vinderea pe piață a obligațiunii). Pentru obligațiunile cu dobândă fixă, mărimea cupoanelor anuale este cunoscută; variația ratei dobânzii pe piață duce la modificarea prețului de piață al obligațiunii; de asemenea, această variație conduce la o altă necunoscută în formula de actualizare a unei sume de bani și anume rata de actualizare (în general se consideră rata medie a dobânzii bancare).

Presupunând rata de actualizare r constantă; pentru o obligațiune perpetuă (fără termen de rambursare) și cu dobândă fixă (fie D această dobândă), valoarea de piață depinde direct de rata de actualizare, astfel:

$$P_0 = D/(1+r) + D/(1+r)^2 + \dots = \sum_{i=1}^{\infty} \frac{D}{(1+r)^i} = \frac{D}{r}$$

În cazul obligațiunilor rambursabile se ajunge la același rezultat:

$$P_0 = D/(1+r) + D/(1+r)^2 + \dots + D/(1+r)^n + P_n/(1+r)^n$$

Presupunem prețul de piață al obligațiunii după n ani neschimbat adică $P = P_n$ și folosind formula pentru însumarea unei progresii geometrice obținem $P_0 = D/r$.

Există obligațiuni la care valoarea de rambursare este mai mare decât valoarea nominală sau valoarea de emisiune este mai mică decât valoarea nominală; în aceste cazuri rentabilitatea actualizată este mai mare decât rata nominală.

3.4.5. Volatilitatea unei obligațiuni

Prețul de piață al obligațiunilor, ca de altfel prețul tuturor valorilor mobiliare, variază datorită tendinței de egalizare a rentabilităților pe diferite tipuri de investiții financiare. Fenomenul de fluctuare a cursului bursier al obligațiunilor în funcție de variația ratei dobânzii pe piață se numește *volatilitatea* obligațiunii. Volatilitatea este procentul de modificare a prețului la o schimbare cu 1% a ratei dobânzii. Diferitele obligațiuni reacționează diferit la variația ratei dobânzii de piață ceea ce conduce la coeficienți diferiți de volatilitate. Volatilitatea este o măsură a sensibilității prețului la schimbările ratei dobânzii pe piață. Când rata dobânzii scade (respectiv crește), prețul obligațiunilor crește (respectiv scade) dar această relație dintre modificarea ratei dobânzii și modificarea prețului nu este una simplă și depinde de caracteristicile interne ale obligațiunii. Volatilitatea este o funcție de:

- Scadență (intervalul de timp rămas până la răscumpărarea obligațiunii); crește o dată cu creșterea scadenței
- Rata cuponului; pentru o scadență dată, volatilitatea crește pe măsura scăderii ratei cuponului
- Nivelul general al ratei dobânzii bancare; pentru o scadență și o rată a cuponului dată, volatilitatea crește o dată cu scăderea nivelului general al ratei dobânzii.

La o obligațiune cu valoarea nominală de 100 u.m., volatilitatea variază astfel:

Obligațiunea	Scadența (ani)	Valoare cupon	Rata dobânzii	Curs de piață	Creșterea prețului
A	10	5	11	64 $\frac{1}{8}$	7,4%
			10	68 $\frac{7}{8}$	
B	20	5	11	51 $\frac{7}{8}$	10,1%
			10	57 $\frac{7}{8}$	
C	10	15	11	123 $\frac{7}{8}$	5,9%
			10	131 $\frac{1}{8}$	

La o obligațiune perpetuă, pornim de la relația $P_0 = C/r$. Prețul obligațiunii urmează funcția $P = f(r)$, funcție ce este continuă și derivabilă. Modificarea prețului de piață al obligațiunii (dP) pentru o variație (dr) a ratei dobânzii pe piață este dată de derivata dP/dr astfel:

$$dP/dr = -C/r^2.$$

Variația prețului obligațiunii în raport cu capitalul investit, dP/P_0 este dată de formula:

$$dP/P_0 = dP/dr \cdot dr/P_0 = -C/r^2 \cdot r dr/C = -dr/r$$

Astfel la o obligațiune perpetuă cu valoarea nominală de 1000 lei, cu o dobândă nominală de 9% în cazul în care rata dobânzii pe piață este de 12% are o volatilitate de:

$$dr = \text{creșterea dobânzii de piață față de cea nominală} = 12\% - 9\% = 3\%$$

$$dP/P_0 = -0,03/0,12 = -25\%.$$

Rezultatul este confirmat și de scăderea cursului bursier al obligațiunii:

$$P_0 = 90/0,12 = 750 \quad dP/P_0 = -250/1000 = -25\%.$$

Fluctuația dobânzii de piață determină o volatilitate substanțială a obligațiunilor în raport cu maturitatea dar în rate descrescătoare.

Dacă se prevede o scădere a ratei dobânzii pe piață, investitorii se orientează spre creșterea volatilității portofoliului de obligațiuni pentru a realiza creșteri cât mai mari ale cursurilor bursiere ale obligațiunilor. În cazul în care se prevede creșterea ratei dobânzii pe piață, investitorii se orientează spre scăderea la maximum a volatilității portofoliului de obligațiuni pentru a se proteja de scăderea cursurilor bursiere a obligațiunilor.

3.4.6. Durata unei obligațiuni (“Duration”)

Duration, un concept dezvoltat pentru prima dată de Frederick Macaulay³⁷, este o caracteristică ce măsoară durata economică a unei obligațiuni și reprezintă măsura maturității obligațiunii.

Acest indicator se calculează măsurând greutatea specifică a valorii actuale a fluxurilor bănești în prețul unei obligațiuni și făcând suma acestora, ponderate cu numărul anilor în cursul cărora are loc plata fluxurilor respective.

$$\text{Duration} = \frac{1}{P} \sum_{t=1}^n \frac{(C_t + D) \cdot t}{(1 + k)^t}$$

unde P este cursul de piață al obligațiunii, C_t este rata scadentă din capital, D este dobânda anual încasată, k factorul de actualizare.

Având în vedere că cursul de piață este o sumă de fluxuri financiare actualizate, *Duration* este de fapt o medie ponderată a timpilor de plată unde ponderile sunt valorile actualizate de plată. Se folosește termenul de *Duration* pentru a diferenția acest indicator de durata obligațiunii care reprezintă intervalul de timp cuprins între momentul emiterii obligațiunii de către emitent și cel al răscumpărării.

Între *Duration* și Volatilitate există relația:

$$\text{Duration} = 100(1 + k) \cdot \text{Volatilitate}.$$

3.4.7. Riscul unei obligațiuni

Riscul e cauza pentru care obligațiunile emise de diverse societăți sunt oferite la o rată nominală mai ridicată decât cea a obligațiunilor guvernamentale. Riscul e absent în cazul celor din urmă; riscul în cazul obligațiunilor emise de firme corporative se manifestă prin eșecul emitentului de a returna sumele către investitori.

O parte integrantă a pieței de capital în general și a pieței obligațiunilor în special o reprezintă agențiile de clasificare a riscului de împrumut denumite agenții de rating. Piața globală a ratingului se extinde continuu datorită utilizării crescânde ca sursă de finanțare a piețelor de capital în locul creditului bancar.

Un *rating* înseamnă o evaluare făcută de o agenție specializată (de exemplu: Moody's Investors' Service, Standard & Poor's Corporation, Fitch Investors' Service Fitch) asupra riscului de neplată la timp și în întregime a principalului și a dobânzii unei obligațiuni de către un debitor. Rating-urile sunt deci niște calificative care se dau în funcție de capacitatea emitentului de a-și achita obligațiile.

Societățile de rating analizează calitatea obligațiunilor și le clasifică în diverse categorii. Pentru investitori aceste evaluări înseamnă o simplificare a sarcinii de evaluare a riscului și de luare a deciziei de investiție. Diverselor categorii de risc le corespund grade simbolizate prin litere. Cel mai înalt grad acordat (Aaa sau AAA) semnifică un risc aproape inexistent. De asemenea, agențiile folosesc și simboluri de + și - pentru a delimita mai precis poziția față de categoria de bază. Prezentăm în continuare coeficienții folosiți de două dintre cele mai importante societăți de rating:

³⁷ Macaulay Duration <http://www.investopedia.com/terms/m/macaulayduration.asp>

Moody's Investors Service	
Aaa	-cea mai bună calitate
Aa	-calitate înaltă
A	-calitate înaltă medie
Baa	-calitate medie
Ba	-posedă elemente speculative
B	-lipsesc caracteristicile unei investiții sigure
Caa	-situație slabă
Ca	-înalt grad speculativ
C	-cele mai riscante

Standard & Poor's Corporation	
AAA	-cel mai înalt grad
AA	-grad înalt
A	-grad mediu-înalt
BBB	-grad mediu
BB	-grad mediu-scăzut
B	-speculativ
CCC-CC	-speculație ridicată
C	-speculație de grad ridicat
DDD-D	-speranță scăzută de recuperare

Un aspect care trebuie analizat este modalitatea în care rating-urile acordate influențează rata dobânzii pe care emitentul trebuie să o plătească pentru a atrage investitorii. Astfel, comparând obligațiuni cu aceeași maturitate, cu cât rating-ul este mai mare cu atât este mai mică rata dobânzii pe care o plătește emitentul. Similar, obligațiunile cu un rating mic trebuie să fie însoțite de dobânzi ridicate, oferind randamente înalte, pentru a convinge investitorii care ar putea fi îngrijorați în legătură cu plata la timp a dobânzii și a principalului.

Nu toate obligațiunile sunt cotate de agenții. Emisiuni mici sau plasamente private în general nu sunt cotate. În general cele două agenții cotează identic obligațiunile; rare sunt cazurile în care să existe un dezacord mai mare de o treptă. Clasificările nu rezolvă în totalitate problemele investitorilor în privința riscului. Ierarhiile se schimbă în timp, în măsura în care apar modificări care să conducă spre acestea.

3.4.8. Concluzii

În urma analizei desfășurate în cadrul acestui capitol, *autorul desprinde următoarele concluzii:*

- Bursa este considerată un barometru extrem de sensibil al unei economii de piață; ea reacționează cu sensibilitate la toate evenimentele, schimbările, evaluările care au loc în lumea de afaceri, în viața social-politică. De aceea, urmărirea cursurilor titlurilor prezintă o importanță aparte.
- Interpretarea bursei ca un reper al lumii de afaceri trebuie să țină seama și de imperfecțiunile acesteia; în anumite situații, bursa poate reflecta un optimism nefondat, o falsă promisiune de câștig, ajungându-se la desprinderea trendului bursier de situația economică reală (în cazul crahului bursier). Totuși, nu în toate cazurile crahul bursier are o fundamentare economică (vezi cazul 1929). Toate acestea apar datorită faptului că tendința bursei este determinată nu numai de anticipări raționale, fundamentate din punct de vedere economic, ci și de reacții subiective. Uneori, optimismul generalizat al investitorilor determină relansarea activității bursiere și economice, tot așa cum pesimismul generalizat al investitorilor poate conduce la stagnarea economică.
- Selecția și gestionarea unui portofoliu format din multitudinea de titluri existente provoacă unui investitor, fie el particular sau instituțional, o serie de probleme insurmontabile fără sprijinul unei abordări științifice, respectiv utilizarea unor modele de selecție și optimizare a portofoliului bazate pe relația rentabilitate-risc. Investitorii experimentați știu că orice decizie de investiție trebuie să aibă o bază solidă de informare, analize, modelare și previziune cu privire la oferta de titluri și evoluțiile lor pe o anumită piață. O astfel de ofertă de titluri, în ultimii ani din ce în ce mai atractivă, provine și de pe piața românească de capital.
- Analiza bursieră face parte, alături de analiza economică și cea financiară, din teoria complexă a piețelor financiare. Această analiză permite stabilirea unui număr de elemente care să fie fundamentul răspunsului la o serie de întrebări ale analistului: care este cea mai sigură investiție, cea mai lipsită de risc, cea mai profitabilă etc. Indiferent de metodele și tehnicile de analiză aplicate, de informațiile de care dispun, de experiență, instinct și de noroc, actorii unei burse de valori urmăresc același scop: obținerea profitului.

- Pentru a surprinde și previziona fluctuațiile bursiere ale valorilor mobiliare este necesară o analiză aprofundată a pieței bursiere, a caracteristicilor interne ale pieței financiare, a structurii portofoliului de titluri, a activității economico-financiare a întreprinderilor emitente, a poziției ramurii industriale în ansamblul economiei naționale, a comportamentului conducătorilor, psihologia acționarilor etc.

CAPITOLUL 4. PROBLEMATICA INTELIGENȚEI ARTIFICIALE

4.1. SISTEME INTELIGENTE

4.1.1. Structura sistemelor inteligente

Volumul datelor utilizate de companii este într-o creștere continuă ajungând astăzi la dimensiuni inimaginabile acum câteva decenii. Indiferent dacă o persoană retrage numerar de la un bancomat, cumpără ceva dintr-un super-market, vinde acțiuni sau intră pe un site Web, datele respective sunt cuprinse în baze de date. Acestea constituie o avere dacă sunt prelucrate și analizate. Sistemele inteligente sunt cele care găsesc tipare și descoperă legături între date. Acestea sunt folosite pe scară largă în primul rând pentru o creștere a calității serviciilor și o reducere a costurilor, numărul domeniilor în care sunt folosite crescând continuu. Sistemele inteligente pot furniza decizii de o mare acuratețe și consistență, le pot furniza mai rapid decât operatorul uman fără a obosi (lucrează 24h/zi, 7 zile/săptămână).

Sistemele inteligente sunt, de fapt, niște sisteme bazate pe cunoștințe, pregătite să utilizeze cunoașterea din cele mai diverse domenii, dacă experții reușesc să o pună la dispoziție și să o redea la momentul oportun sau să o utilizeze în raționamente specifice. Ele constituie, la ora actuală, o tehnologie informațională de vârf, în continuă dezvoltare.

În categoria sistemelor inteligente se înscriu **rețelele neuronale, algoritmi genetici, sistemele fuzzy, sistemele expert și sistemele hibride.**

4.1.2. Caracteristici ale sistemelor inteligente

Caracteristicile cele mai importante ale sistemelor inteligente, care le fac atractive considerăm că sunt următoarele:

a) Capacitatea de a lua decizii – reprezintă cea mai importantă trăsătură a sistemelor inteligente aplicate în afaceri și constă în abilitatea de a lua decizii pe baza datelor de care dispun și a modelelor create ca urmare a miilor sau sutelor de tranzacții anterioare. Aceste sisteme se dezvoltă în timp pe baza multor ani de experiență în tratarea sarcinilor de afaceri.

b) Adaptarea – este capacitatea unui sistem inteligent de a-și putea revizui cunoștințele de care are nevoie în funcție de noile modificări survenite în mediu. Afacerile sunt în continuă mișcare și ca urmare sistemele ce le tratează trebuie să se actualizeze rapid, să includă schimbările. Tiparele în modul de efectuare a cheltuielilor folosind cărțile de credit sunt semnificativ diferite pe timpul unei recesiuni comparativ cu cazul creșterii economice. Sistemele inteligente au capacitatea de a se adapta acestor schimbări mai ales în mediul afacerilor.

c) Flexibilitatea – reprezintă capacitatea sistemelor inteligente de a lua decizii cu informații incomplete, și de a recunoaște diferite șabloane în diverse condiții neîntâlnite până în prezent. Deciziile umane sunt în general flexibile. Omul poate lua decizii chiar dacă informațiile de care dispune sunt imprecise și incomplete.

d) Explicarea, interpretarea – sistemele inteligente au potențialul automatizării deciziilor, dar deciziile trebuie să fie înțelese, să fie “transparente”. În timp ce sistemele expert realizează o explicare completă, rețelele neuronale au dificultăți în această privință. Este necesar să se explice deciziile căci numai așa se pot corecta erorile, se pot îmbunătăți deciziile luate. Transparența permite o interacțiune mai puternică cu experții umani ceea ce duce la creșterea calității.

e) Descoperirea, învățarea – constau în faptul că sistemele inteligente nu pot numai automatiza sarcini ci au de asemenea posibilitatea descoperirii de noi procese și relații, lărgindu-se astfel baza de date.

4.1.3. Etape în rezolvarea unei probleme folosind sisteme inteligente

Rezolvarea unei probleme practice presupune urmărirea unei anumite căi (“*Problem solving path*”) mai mult sau mai puțin standardizată. Aplicațiile clasice urmează o serie de etape standardizate de proiectare (stabilire situații de intrare-ieșire, structurarea bazei informaționale în subscheme, stabilire unități funcționale și de prelucrare etc). În cazul aplicațiilor inteligente nu există astfel de rețete generale datorită creativității care caracterizează acest domeniu.

Kevin Knight³⁸ a enunțat patru etape de principiu în rezolvarea unei probleme folosind inteligența artificială:

- 1) Definirea precisă a problemei
- 2) Analiza problemei
- 3) Reprezentarea cunoștințelor
- 4) Alegerea celei mai bune tehnici de rezolvare

Parcursul integral al acestor etape nu este obligatoriu.

Definirea problemei implică căutarea printr-un spațiu (mulțime) de stări ("State space search") și presupune parcurgerea următoarelor subetape:

- Definirea spațiului de stări
- Specificarea unei stări inițiale, adică descrierea a una sau mai multe stări din care se poate începe rezolvarea problemei ("Initial states")
- Specificarea uneia sau mai multor stări care constituie o bună rezolvare a problemei ("Goal states")
- Specificarea unui set de reguli care descriu acțiunile permise.

Mișcarea prin mulțimea stărilor (căutarea) se face aplicând regulile specificate într-o anumită ordine, generând astfel o strategie de control ("Control Strategy"). Controlul de strategie este operațional când mai multe reguli sunt potrivite pentru următoarea mișcare. Un bun control al strategiei creează mișcare prin spațiul stărilor și este sistematic. Dezavantajul este explozia combinatorică în momentul încercării evaluării tuturor căilor de rezolvare a problemei. Consumul mare de timp și resurse conduce spre o altă abordare de căutare prin spațiul soluțiilor și anume căutarea euristică. Spre deosebire de căutarea sistematică totală, căutarea euristică alege pas cu pas cea mai bună alternativă în rezolvarea problemei. Aceasta conduce spre o altă definiție a inteligenței artificiale: "Studiul tehnicilor de rezolvare a problemelor exponențiale în timp polinomial prin exploatarea cunoștințelor despre domeniul problemei".

Tehnicile euristice definite de Kevin Knight sunt:

- Căutarea întâi în adâncime "Depth First Search"
- Căutarea întâi în lățime "Breadth First Search"
- "Hill Climbing"
- "Best First Search"
- "Problem Reduction"
- "Constraint Satisfaction"

4.1.4. Rețelele neuronale

Colaborarea dintre un neurofiziolog și un matematician (Warren McCulloch și Walter Pitts) a marcat în anul 1943 apariția primului model de neuron artificial. Acest model presupune că neuronul funcționează ca un dispozitiv simplu ale cărui intrări sunt ponderate. Ponderile pozitive sunt excitatoare, iar cele negative inhibitoare. Dacă excitația totală, adică suma ponderată a intrărilor depășește un anumit prag, atunci neuronul este activat și emite semnal de ieșire (convențional valoarea este notată cu 1). Dacă această valoare se găsește sub prag, neuronul nu este activat și ieșirea lui se consideră a fi 0. Funcția asociată modelului McCulloch-Pitts, numită și funcție prag este următoarea:

$$f: \mathbb{R} \rightarrow \{0, 1\} \quad \begin{aligned} f(x) &= 1 \text{ dacă } x \geq 0 \\ f(x) &= 0 \text{ dacă } x < 0 \end{aligned}$$

Unitatea de organizare a sistemului nervos este neuronul, o celulă care prezintă un număr de dendrite și un axon, prin intermediul cărora se interconectează cu alți neuroni. Dendritele constituie intrările în celula neuronală, iar axonul ieșirea. Axonul se ramifică astfel încât neuronul poate conecta ieșirea la mai mulți neuroni. Totalitatea impulsurilor primite la intrarea neuronului îl pot excita astfel încât să genereze un impuls mai departe spre neuronii cu care este conectat. Legăturile dintre neuroni sunt ponderate și fiecare neuron aplică o transformare asupra impulsului de la intrări înainte de a-l transmite mai departe.

³⁸ [Knight90] Knight K., Rich E. –Artificial Intelligence - McGraw-Hill, 1990

Hebb a propus în 1949 un mecanism calitativ ce descrie procesul prin care conexiunile sinaptice sunt modificate pentru a reflecta mecanismul de învățare realizat de neuronii interconectați atunci când aceștia sunt influențați de anumiți stimuli ai mediului.

Rosemblatt în 1958 a propus un dispozitiv numit perceptron, care este bazat pe interconectarea unei mulțimi de neuroni artificiali. Acesta este primul model de rețea neuronală artificială.

În 1969 Minski și Papert au demonstrat că toate modelele simple de tipul perceptronului prezintă numeroase limitări. Astfel, un perceptron elementar nu poate folosi funcția logică “sau exclusiv” ceea ce a condus la scăderea interesului pentru rețelele neuronale pentru mulți ani

Rețelele neuronale au reapărut în anii '80 modelând soluțiile problemelor prin instruirea neuronilor artificiali conectați în rețele configurate în prealabil cu ajutorul unor metode de dezvoltare specifice. Ele funcționează pe principiul celulelor creierului uman. Acestea învață să recunoască comportamente din exemple, chiar dacă sunt inexacte și incomplete. Ele sunt utilizate în probleme de previzionare utilizând metode de regresie liniară și servicii care au fost îmbunătățite de experți după mulți ani de experiențe.

Rețelele neuronale artificiale sunt alcătuite dintr-o mulțime de noduri în care se află neuronii artificiali, elemente de procesare neliniare, care operează în paralel. Neuronul artificial este o copie mult simplificată a neuronului biologic. Acesta este format dintr-un corp, un set de intrări și o ieșire. Fiecare intrare este ponderată, adică valoarea sa este multiplicată cu o valoare corespunzătoare, numită pondere, apoi toate intrările ponderate sunt însumate. Fiecare dintre neuroni realizează o operație simplă și comunică rezultatele unităților învecinate. Nodurile rețelei sunt organizate pe nivele care comunică cu nodurile nivelelor următoare. Fiecărei legături i se asociază o pondere și o valoare. În timpul recunoașterii comportamentelor, fiecare nod operează ca un dispozitiv de intrare: adună toate intrările și aplică funcții. În rețea conexiunea este foarte puternică, permițându-se construirea de arhitecturi multiple. Un algoritm de învățare este o procedură care e utilizată pentru a găsi valori ale ponderilor pentru fiecare sarcină. O intrare este prelucrată și determină o ieșire la nivelul de ieșire. Ieșirea este comparată cu un șablon ideal și eroarea este propagată înapoi în rețea pentru a se face corecția și ciclul se repetă până ce eroarea a fost eliminată sau e foarte mică.

Se cunosc mai mulți algoritmi de învățare a rețelelor neuronale, secretul lor fiind flexibilitatea și generalizarea. Rețelele neuronale sunt foarte bine folosite pentru sistemele neliniare. Acestea le dă un avantaj asupra multor metode statistice utilizate în afaceri și finanțe. De asemenea sunt foarte eficiente pentru învățarea comportamentelor pentru date incomplete, zgometoase, unde experții nu sunt disponibili, unde regulile clare nu pot fi ușor formulate.

Una din caracteristicile cele mai importante ale rețelelor neuronale este capacitatea de a învăța. Odată ce o rețea a învățat un comportament, un mod de rezolvare a unei sarcini, ea va ști să ia decizii în condiții similare, chiar dacă persoana respectivă mai prezintă sau nu datele referitoare la ea. Limitarea lor constă în posibilitatea redusă de explicare a capabilităților, concluziile neputând fi explicate. Reprezentarea cunoștințelor prin rețele neuronale are un mare dezavantaj: cunoștințele sunt total opace. Din acest motiv, un doemniu major de interes în cercetarea rețelelor neuronale îl constituie dezvoltarea regulilor pentru reprezentarea procesului folosit de rețelele neuronale.

Există numeroase sisteme neuronale în sectorul comercial și financiar-bancar dintre care menționăm:

VISA INTERNATIONAL are un sistem bazat pe o rețea neuronală de detectare a fraudelor. Ea este antrenată pentru a verifica activitățile frauduloase prin compararea utilizărilor cărților de credit cu cunoscute cazuri de fraudă. Prin verificarea problemelor ca cea a momentului tranzacției și tipul activității pentru fiecare folosire a cărții de credit VISA, rețeaua poate semna alarma în cazul unor potențiale abuzuri.

CHRIS –Cardholder Risk Identification Service asigură validitatea cărților de credit automat la pompele de benzină. Rețeaua neuronală este utilizată pentru a verifica tranzacțiile și de a le raporta pe o scară de la 0 la 999 cu un scor ridicat de indicare a șabloanelor de fraudă.

AAA, AAB sunt sisteme de predicție a falimentului băncilor sau de previziune a clasificărilor titlurilor de credit emise de o societate comercială pe scară prestabilită.

Fujitsu a prezentat o aplicație cu rețele neuronale pentru previziuni pe piața valorilor mobiliare. Aceste rețele au fost implementate de Nikko Securities dar sunt disponibile și pentru alți brokeri.

4.1.5. Algoritmii genetici

Termenul de *algoritm genetic* a fost folosit pentru prima dată în 1967 de către J.D. Bagley. Utilizarea efectivă a apărut mai târziu, în 1975 când John Holland a prezentat o metodă de optimizare bazată pe principiile de evoluție genetică.

Caracteristicile esențiale ale procesului de evoluție genetică sunt:

- Cromozomii sunt purtătorii informației genetice
- Fiecare individ posedă un număr determinat de cromozomi. Totalitatea acestora reprezintă genotipul său.
- Cromozomii sunt structuri liniare alcătuite din gene. Genele poartă caracteristicile ereditare. O genă controlează una sau mai multe caracteristici.
- Evoluția este un proces ce operează la nivelul cromozomilor.
- Selecția naturală reprezintă legătura dintre cromozomi și performanțele indivizilor. Procesul selecției naturale favorizează reproducerea acelor cromozomi ce codifică structuri de succes.
- Evoluția se realizează în procesul reproducerii. În evoluție acționează procese de selecție și mutație. Foarte importante sunt procesele de recombinare a materialului genetic ce caracterizează părinții.

Algoritmii genetici se inspiră din evoluțiile biologice bazate pe sistemul darwinist: supraviețuirea celor mai puternici. Ei reprezintă o clasă de algoritmi evolutivi. Algoritmii evolutivi implementează proceduri care imită procesele de adaptare și căutare apărute în evoluția naturală.

Sistemele evolute se obțin după mai multe cicluri, fiecare ciclu desemnându-și propria generație și alegându-și soluțiile candidat care se adaptează cel mai bine problemei. Soluțiile candidat contribuie la rezolvarea unei probleme, iar cele ce nu dau rezultatul dorit sunt penalizate. Se începe cu o populație de soluții pentru o anumită problemă și se produc noi generații tot mai performante printr-o varietate de scheme: șiruri binare, numere reale, reguli.

Structura generală a unei proceduri evolutive arată astfel:

P1. Se inițializează populația $P(t)$

P2. Se evaluează $P(t)$ utilizând o funcție de performanță

P3. Până când o condiție de oprire devine adevărată se execută

{ $t := t + 1$

Selecția $P(t)$

Recombinarea asupra lui $P(t)$

Mutația asupra lui $P(t)$

Evaluarea lui $P(t)$

Supraviețuirea lui $P(t)$

}

Conform acestei proceduri, pașii generali care trebuie urmați în construirea unui sistem inteligent bazat pe algoritmi genetici sunt:

- crearea unei populații inițiale (soluții candidat la rezolvarea unei probleme)
- evaluarea fiecărui membru din această populație folosind funcții de evaluare
- selecția celor mai promițători membri ai populației inițiale, cei care s-au adaptat cel mai bine necesităților problemei de soluționat
- construirea de noi membri din cei existenți care să aibă caracteristicile cele mai bune. Această operație se face prin reproducere, folosind operatori genetici de încrucișare și mutație pentru a obține noi membri.
- evaluarea gradului în care noii membri corespund mai bine soluționării problemei, caz în care se abandonează vechea populație și se înlocuiește cu populația din noua generație.
- verificarea unei condiții de terminare a procesului, care dacă nu se respectă se revine la pasul 2.

Aceste sisteme inteligente au fost descrise în anii '70 de John Holland. Algoritmii genetici se pot ușor adapta schimbărilor și pot fi utilizate ca instrument pentru descoperirea unor cunoștințe noi, chiar pornind de la date incomplete. Se folosesc acolo unde nu este sigură metoda prin care se rezolvă o problemă, problemă care are un spațiu foarte mare de căutare a soluțiilor. Cele mai bune utilizări ale algoritmilor genetici sunt în căutarea unor soluții și metode neliniare care nu au mai fost dezvoltate

matematic. Ei se dovedesc eficienți în probleme de căutare și identificare a structurilor și relațiilor specifice în cadrul bazelor de date și de cunoștințe voluminoase. Un succes deosebit a fost obținut în probleme de optimizare referitoare la selecția personalului și a portofoliilor precum și la predicția falimentului întreprinderilor, pe baza indicatorilor financiari obținuți din bilanț.

Utilizarea algoritmilor genetici a luat amploare și în domeniul acordării creditelor. Un astfel de exemplu de aplicație este cea de acordare a creditelor pornind de la reguli aleatoare. La fiecare generație se verifică cât de bine rezolvă regulile problema. O regulă care face mai puține erori în clasificarea acordărilor de credite va avea o supraviețuire mai lungă decât cea care face mai multe erori. Construirea părților unei reguli se bazează pe alte reguli anterioare. Cu cât regulile construite satisfac problema cu atât au șanse de supraviețuire în noua generație.

4.1.6. Sistemele fuzzy

Mulțimile fuzzy și logica fuzzy, ale cărei baze au fost puse de Lofti Zadeh (1965), profesor la Berkley University of California, sunt definite ca instrumente pentru lucrul cu incertitudini non-probabilistice. Ambiguitatea inerentă a unor fenomene sau a unor concepte ale limbajului natural nu poate fi redusă la o măsură probabilistică.

Primele idei care au condus spre logica fuzzy pornesc încă de la Platon, care a indicat faptul că există o a treia regiune între adevărat și fals. Apoi Lukasiewicz a fost cel care a propus o alternativă sistematică la logica bivalentă a lui Aristotel. Prin teoria logicii trivalente, el a propus o a treia valoare de adevăr, tradusă prin “posibil”, căreia i-a asignat o valoare numerică între adevăr (1) și fals (0). Mai târziu el a explorat logicile cu patru și cu cinci valori de adevăr și a declarat ulterior că nu există nimic care să împiedice obținerea logicii cu un număr infinit de valori.

Logica fuzzy oferă posibilitatea de a modela condițiile care sunt inerent imprecis definite. Caracterul vag (fuzzy) din procesele de gândire umană sugerează faptul că multe din aspectele logice ale procesului de gândire uman nu au la bază logica cu două valori sau cu valori multiple ci al logicii cu adevăruri fuzzy. Asocierea unor semnificații lingvistice, a unui set de termeni, pentru un set de valori numerice, obținând astfel un set de mărimi fuzzy poartă numele de *fuzzyficare*. Procesul invers, care asociază unei mărimi fuzzy o valoare numerică precisă poartă numele de *defuzzyficare*. Prezentăm în figura 14 fluxul acestor procese.

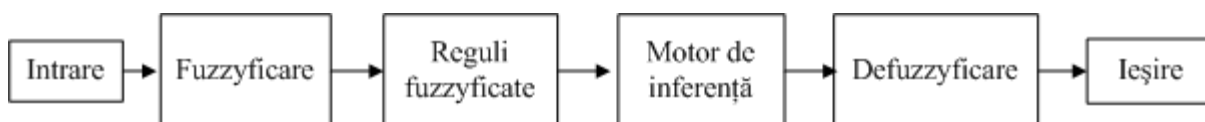


Fig. 14 Procesele de fuzzyficare și defuzzyficare

Cele două procese sunt specifice tuturor aplicațiilor de logică fuzzy. Aceste procese trebuie definite astfel încât această dublă translație dintr-un plan în altul să nu altereze nici coerența sistemului logic calitativ nici coerența sistemului fizic numeric. Raționamentul fuzzy este procesul de inferare a concluziilor dintr-un set de reguli fuzzy care acționează asupra cunoașterii fuzzyficate.

Mulțimile fuzzy pot modela limbajul imprecis cu concepte ca: mic, mare, frumos, urât, ridicat, scăzut. Incertitudinea cu care tratează mulțimile fuzzy permite procesarea unor valori neambigue pentru controlul cunoașterii ambigue. În contrast cu sistemele bazate pe reguli convenționale în care o singură regulă e activă pentru a da un răspuns când o condiție este adevărată, în sistemele fuzzy toate regulile contribuie la rezultatul final (fiind vorba de un proces de agregare). Unele din avantajele sistemelor fuzzy sunt posibilitatea de a trata date imprecise și faptul că regulile pot fi actualizate corect. Bazele de cunoștințe folosesc structuri de tip reguli de producție foarte ușor de examinat, înțeles sau modificat. Principala regulă de inferență este Modus Ponens care înseamnă utilizarea unei reguli pentru a infera o concluzie dintr-o premiză: Dacă A este adevărat și A implică B atunci și B este adevărat. Dezavantajul sistemelor fuzzy este că funcțiile membru și regulile trebuie specificate manual ceea ce duce la consum mare de timp și erori. Sistemele fuzzy nu sunt așa de ușor adaptabile la schimbările mediului. Se înregistrează o creștere a numărului de aplicații fuzzy “database retrieval” deoarece regăsirea în bazele de date convenționale nu oferă o foarte bună flexibilitate operatorului uman.

Autorul consideră că tehnicile fuzzy în forma raționamentului aproximativ furnizează, pentru asistarea deciziilor și pentru sistemele expert, puternice capacități de raționament. În economie, teoria fuzzy are implicații profunde în modelarea fenomenelor financiar-bancare, luarea deciziilor multicriteriale, știința managementului și marketingului. Sistemele fuzzy dispun de o flexibilitate deosebită în aplicații de control industrial, recunoașterea structurilor (scrisul de mână) precum și aplicații de evaluarea creditelor.

Fuji Bank din Tokyo a realizat un sistem fuzzy pentru comerțul cu titlurile de credit pe termen scurt; este utilizat pentru operații zilnice și conține aproximativ 2000 de reguli. Este operațional din 1993 și aduce câștiguri de aproximativ 770.000 USD lunar. Japonezii au creat o inginerie fuzzy cu mult succes prin soluții dezvoltate de Laboratorul Internațional de Inginerie Fuzzy din Yokohama.

4.1.7. Sistemele expert

Dacă toate cunoștințele ar putea fi înglobate într-un formalism, nu ar mai fi nevoie de imaginarea unor tehnici de programare bazate pe cunoștințe. Există numeroase domenii în care nu există un algoritm sau algoritmul de rezolvare este inacceptabil din punct de vedere al timpului de execuție. De asemenea în practică ne întâlnim cu domenii incomplet specificate sau cu o specificație care se modifică des. Pentru aceste domenii există specialiști, experți care pot rezolva probleme complexe, având la bază experiența pe care o dețin. Pornind de la aceasta, s-a ajuns la ideea de a dezvolta așa numitele *sisteme expert* care să poată rezolva probleme la nivelul experților umani într-un anumit domeniu. Pentru dezvoltarea de aplicații în acest sens, s-a plecat de la analiza modului în care un expert rezolvă o astfel de problemă. În urma încercărilor de a simula inteligența umană, s-a ajuns la concluzia că o caracteristică a experților umani este că dețin un bagaj considerabil de cunoștințe în domeniul respectiv. Cu aceste cunoștințe se fac operații de abstractizare, generalizare, particularizare, clasificare etc. Astfel s-a ajuns la ideea că pentru rezolvarea unor astfel de probleme se impune reprezentarea explicită a cunoștințelor domeniului respectiv.

Sistemele expert reprezintă tehnologia cea mai veche și mai consacrată dintre sistemele inteligente. Acestea presupun stocarea cunoștințelor expertului uman în programe pe calculator care permit manipularea lor ulterioară de către calculator.

Puterea sistemelor expert constă în motorul de inferență și sistemul de explicații de care dispun. Un sistem expert puternic are un modul explicit de reprezentare a cunoștințelor astfel încât acestea să fie ușor de citit și înțeles, să poată da explicații despre modul cum s-a ajuns la o anumită decizie. Dezavantajul sistemelor expert este că nu sunt capabile să învețe, să se autoinstruiască și deci nu se pot adapta ușor la modificările mediului cu care operează.

Yamaichi Securities a realizat un sistem expert integrat ce este util pentru securitatea investițiilor. Banca de cunoștințe constă în cazuri de investiții sigure, know-how-ul din marile case de brokeraj și alte informații de securitate colectate și procesate prin analize astfel încât sistemul să poată semnaliza alarme în cazul unor modificări dubioase pe piață. Regulile acestui sistem expert operează pe patru niveluri:

- macroreguli –aleg ramura și efectul de comerț
- reguli industriale –cercetarea indicatorilor industriali
- microreguli –cercetarea anumitor titluri de valoare individuale
- regulile investitorilor –preferințele investitorilor

Pentru a administra capitalul moștenit, IBM a dezvoltat un sistem expert: Capital Asset Expert System Cases care oferă consultații, dă utilizatorului informații și utilizează proceduri și forme din mai multe surse. Sunt create proceduri și directive făcând acest sistem disponibil 24 h / zi, 7 zile / săptămână.

Sistemele expert urmează a fi prezentate mai în amănunt în capitolul 4.3.

4.1.8. Agenții inteligenți

Agenții inteligenți software sunt programe complexe, cu o oarecare perenitate, care execută sarcini specifice în contul și în beneficiul unui utilizator și posedă inteligență care le permite să efectueze sarcini în mod autonom și să interacționeze într-un mediu de rețea. Noțiunea de “agent” este relativ nouă.

Wooldridge și Jennings³⁹ definesc un agent inteligent ca un sistem hardware sau software cu următoarele proprietăți:

- *autonomie* –un agent operează ca un proces de sine stătător, fără intervenție umană directă și deține controlul asupra acțiunilor și stării sale interne
- *reactivitate* –agentul percepe mediul în care acționează și reacționează prompt la schimbările petrecute în mediul respectiv
- *proactivitate* –această caracteristică se referă la capacitatea agentului de a manifesta un comportament orientat spre scop prin preluarea inițiativei
- *abilitate socială* –agentul interacționează cu alți agenți (sau cu oamenii) printr-un anumit limbaj de comunicare.

Un agent inteligent este înzestrat cu cunoștințe în ceea ce privește dorințele utilizatorului, utilizând aceste cunoștințe pentru a-și executa sarcinile. Un program tradițional poate fi considerat un agent non-inteligent deoarece execută sarcini specifice, ușurând munca utilizatorului. Spre deosebire de acestea, un agent inteligent execută sarcinile în mod autonom, cerând intervenția utilizatorului doar pentru decizii importante.

Un agent interacționează cu mediul pentru a-și atinge scopul. El adună informații din mediul său de lucru și ia decizii bazate pe aceste informații. Caracteristicile unui agent inteligent pot fi clasificate astfel: interne (abilitatea de a învăța, reactivitatea, autonomia și scopul orientat) și externe (comunicarea și cooperarea).

În funcție de scopul pentru care au fost creați, agenții inteligenți se clasifică în trei categorii:

- *agenți de informare* –sarcina lor este de a localiza și extrage sursele de informație, filtrând partea necesară utilizatorului și prezentând apoi rezultatele utilizatorului
- *agenți de cooperare* –au sarcina de a rezolva probleme complexe folosind mecanisme de cooperare și comunicare cu alte obiecte
- *agenți tranzacționali* –rolul lor este de procesare și monitorizare a tranzacțiilor

Din punct de vedere al mobilității, agenții se clasifică în două categorii:

- *statici* –acești agenți nu își părăsesc locul și deci nu se deplasează prin rețea
- *dinamici* –agenții dinamici se deplasează prin rețea pentru a-și îndeplini sarcinile pentru care au fost proiectați.

În ceea ce privește limbajul în care se pot implementa agenții inteligenți, se constată că acesta trebuie să satisfacă anumite cerințe: orientare pe obiecte (agenții posedă date și metode), independența de platformă, securitate, posibilități de manipulare a codului (limbajul trebuie să aibă mecanisme care să identifice codul agentului de alte obiecte sau să poată transfera codul programului în rețea). Datorită acestor caracteristici, mediul cel mai utilizat pentru dezvoltarea agenților inteligenți este cel bazat pe Java.

Evoluția deosebită pe care o are această tehnologie (numită de numeroase autorități științifice și firme din domeniul informaticii “tehnologie cheie”) a fost favorizată și accelerată de dezvoltarea extraordinară a Internetului și de complexitatea crescută și necesitățile specifice ale noilor activități de tip “e-Business”.

În prezent există o multitudine de agenți inteligenți funcționali. Majoritatea motoarelor de căutare dispun de agenți inteligenți care caută informații pe web, le filtrează și apoi le clasifică prin diverse procedee de indexare, permițând apoi regăsirea rapidă a lor. În domeniul comerțului electronic, există agenți care sprijină cumpărarea și vânzarea de produse și servicii pe Internet pentru utilizatorii săi.

Tehnologia agenților inteligenți este o mare promisiune pentru viitorul societății informaționale. Există însă întrebări încă deschise cu privire la modalitățile de interacțiune cu alți agenți precum și la siguranța că un agent va asculta și se va conforma dorințelor utilizatorului, asigurând și securitatea datelor.

4.1.9. Sistemele hibride

³⁹ [Wooldridge95] Wooldridge M, Jennings N.R. “Agent Theories, Architectures, and Languages: a Survey”, in Wooldridge and Jennings Eds., Intelligent Agents, Berlin, Springer-Verlag, 1995

Hibridizarea se referă la funcționări realizate cu ajutorul unei tehnologii încorporate într-un sistem inteligent dezvoltat cu o anumită tehnologie inițială, considerată tehnologie de bază la care se adaugă și alte tehnologii pentru a profita la maximum de avantajele de care dispun diversele tipuri de sisteme inteligente.

Diferitele componente ale sistemelor hibride comunică rezultatele uneia sau alteia, obținându-se rezultatul final. Aceste sisteme acoperă nu numai simpla combinare a diferitelor tehnologii ci și integrarea tehnologiilor inteligente cu sisteme clasice, conținând baze de date sau calcul tabelar. Multiplele avantaje ale sistemelor hibride pe care teoria și practica le-au relevat deja sunt următoarele:

- folosirea informațiilor din mai multe surse
- tezaurizarea informațiilor strategice pentru utilizarea lor ulterior în mod eficient
- obținerea unor modele de sisteme hibride natural în care diferitele tehnologii sunt definite ca obiect.

De exemplu, combinarea algoritmilor genetici cu sisteme expert se poate face în două moduri:

- soluțiile sistemului expert vor constitui populația inițială a unui algoritm genetic
- algoritmul genetic va genera parametri folosiți de sistemul expert în regulile de producție sau ca factori de încredere.

Uneori procesul de antrenare a unei rețele neuronale rămâne “prins” într-un minim local, ponderile rețelei atingând un nivel relativ bun dar nu optim. În aceste cazuri se impune crearea unui sistem hibrid folosind algoritmi genetici în scopul evitării minimelor locale. Eficacitatea dovedită de algoritmi genetici în procesul de învățare a rețelelor neuronale a determinat încorporarea acestora în numeroase produse program profesionale.

În domeniul planificărilor financiare și elaborării bugetelor unei întreprinderi se pot identifica trei tipuri de probleme:

1. predicția –rezolvată cu sisteme neuronale
2. optimizarea –rezolvată cu algoritmi genetici
3. raționamentul secvențial –rezolvat cu sisteme expert

Combinăția dintre rețelele neuronale și sisteme fuzzy se poate utiliza cu succes în probleme decizionale unde datele de intrare au caracter incert sau incomplet. Asupra datelor se aplică un sistem de preprocesare de tip fuzzy după care are loc o transformare a datelor astfel încât să fie recunoscute de rețeaua neuronală. Atribuind scoruri pentru învățare, sistemele fuzzy primesc un punctaj slab, iar rețelele neuronale un punctaj bun.

Un sistem hibrid bazat pe sisteme expert și rețele neuronale combină avantajele celor două tipuri de tehnologii obținându-se aplicații performante din punct de vedere al adaptabilității, vitezei de procesare, fiabilității și interfeței cu utilizatorul. La sistemele expert, crearea bazei de cunoștințe este foarte mare consumatoare de timp. O schimbare în baza de cunoștințe poate implica un întreg proces de modificări pentru a aduce sistemul în parametri normali de funcționare datorită multiplelor interdependențe existente între entitățile bazei de cunoștințe. În cazul rețelelor neuronale modificarea mediului de lucru nu implică decât reantrenarea rețelei pe baza noilor seturi de date. Aceste caracteristici fac ca sistemele expert să se focalizeze pe domenii preponderent statice, mediile dinamice cum ar fi cel economic orientându-se mai mult spre rețelele neuronale, datorită acestei caracteristici de autoorganizare și funcționare fără o bază de cunoștințe programată a acestora din urmă. Un alt avantaj al sistemelor neuronale constă în utilizarea datelor eronate (pline de zgomote informaționale), ceea ce este mai dificil pentru sistemele expert. În schimb avantajul net al sistemelor expert constă în explicarea rezultatelor. Combinarea avantajelor celor două tipuri de sisteme inteligente poate conduce spre rezultate remarcabile. Astfel, un sistem hibrid poate partaja o bază de cunoștințe în care sistemul expert generează explicațiile, iar rețeaua neuronală dă soluția problemei.

Bursa de Valori din Londra a realizat un sistem inteligent hibrid pentru detectarea afacerilor dubioase în care se combină algoritmi genetici cu logica fuzzy și cu rețelele neuronale în scopul supravegherii efectuării tranzacțiilor zilnice.

Irish Bank a dezvoltat un sistem de asistare a deciziilor care alocă fondurile și tratează emisiunile de obligațiuni pentru a-și optimiza activitatea. Garanțiile sunt cercetate cu algoritmi euristici, rezultând un sistem hibrid pentru protejarea clienților.

Nikko Securities a realizat un sistem expert OTT utilizat pentru tranzacționare. Ulterior, acest sistem a fost convertit într-un sistem suport de cunoștințe, un asistent de determinare a noilor instrumente, bazat pe rețele neuronale și sisteme fuzzy, devenind un sistem hibrid.

4.2. SISTEME DE ASISTARE A DECIZIILOR

4.2.1. Noțiuni despre procesul decizional

4.2.1.1. Noțiuni introductive

Procesul decizional circumscrie totalitatea procedurilor utilizate de către decident în rezolvarea problemei decizionale cu care acesta se confruntă. În raport cu gradul de detaliere a etapelor pe care le implică procesul decizional, putem evidenția trei niveluri ale acestuia⁴⁰:

- *Nivelul organizațional* cuprinde etapele cele mai generale ale procesului decizional, cum ar fi: pregătirea și analiza datelor, pregătirea problemei decizionale, elaborarea alternativelor etc.
- *Nivelul informațional* detaliază etapele nivelului anterior în subetape. Acestea au caracter informațional deci presupun culegerea, stocarea, prelucrarea și evaluarea informației.
- *Nivelul tehnologic* include activități omogene din cadrul procesului decizional, cum ar fi proceduri și operații algoritmice și euristice dar și din operații logice care au loc în mintea decidentului.

Procesul decizional poate fi privit ca o succesiune de activități desfășurate de decident (individual sau de grup) cu scopul de a alege varianta optimă din variantele posibile de a acționa. În orice proces decizional de management se regăsesc următoarele elemente:

- *Decidentul* este persoana sau grupul de persoane care este mandatat să aleagă varianta optimă conform criteriilor decizionale utilizate.
- *Problema decizională* -Decizia are drept scop soluționarea unei probleme decizionale. În absența problemei, decizia nu are obiect.
- *Mulțimea variantelor decizionale* poate fi finită sau infinită. Aceasta cuprinde totalitatea posibilităților de soluționare a problemei decizionale. Definirea acestei mulțimi necesită informații din interiorul și din afara organizației, gândire managerială creatoare, consultarea experților. După definirea problemei decizionale are loc identificarea alternativelor decizionale.
- *Mulțimea criteriilor decizionale* include o serie de caracteristici pe baza cărora se evaluează și compară variantele decizionale, în vederea adoptării celei mai eficiente decizii. Mulțimea criteriilor decizionale este reprezentată de punctele de vedere ale decidentului care izolează aspecte ale realității economice în cadrul procesului decizional.
- *Mediul ambiant* (condițiile obiective) este reprezentat de ansamblul condițiilor interne și externe care sunt influențate și influențează decizia. În mediul ambiant se pot manifesta, pentru o anumită situație decizională, mai multe stări ale condițiilor obiective. În general, mediul ambiant se caracterizează printr-o mobilitate deosebită. Evoluția acestuia se manifestă uneori contradictoriu în ceea ce privește influența asupra procesului de luare a deciziilor în cadrul firmei. Astfel, creșterea complexității activității firmei influențează în mod nefavorabil fundamentarea deciziilor, în timp ce ridicarea nivelului de pregătire profesională și în domeniul managementului are efecte favorabile asupra luării deciziilor.
- *Mulțimea consecințelor* cuprinde ansamblul rezultatelor ce s-ar obține aplicând, pe rând, fiecare variantă decizională, conform fiecărui criteriu decizional și fiecărei stări a condițiilor obiective.
- *Obiectivele deciziei* sunt nivelele propuse de către decident pentru a fi atinse în urma implementării variantei decizionale alese.
- *Utilitatea* fiecărei consecințe aferente diferitelor variante decizionale este dat de efectul așteptat de decident în urma faptului că o anumită consecință se realizează.

⁴⁰ [Mărăcine98] Mărăcine V. –Decizii manageriale, Editura Economică, București, 1998

Managementul este procesul prin care sunt atinse anumite scopuri folosind anumite resurse considerate intrări. Scopurile atinse constituie ieșiri ale procesului decizional. Toate activitățile implică luarea unor decizii. În prezent aceste decizii sunt din ce în ce mai greu de luat datorită pe de o parte creșterii complexității mediului, a numărului extrem de mare de alternative, a incertitudinii în creștere iar pe de altă parte costului posibilelor erori cauzate de aceste decizii care pot fi uneori fatale în mediul concurențial de azi. De aceea deciziile nu pot fi luate după principiul încercărilor, managerii trebuie să-și diversifice metodele și să învețe să folosească noile unelte și tehnici care apar. În acest domeniu, gradul ridicat de incertitudine al viitorului necesită utilizarea unor instrumente și tehnici care folosesc analiza cantitativă, având ca suport rezultate de vârf ale informaticii. Toate se constituie într-o nouă știință a managementului.

Sistemul managerial al unei firme poate oferi o imagine reprezentativă a ierarhizării necesităților informaționale la diferite nivele, astfel:

- nivelul 1 –funcționarii, managerii I –lucrează cu informații structurate. La acest nivel cerințele informaționale se asigură pe baza tehnicilor soft clasice.
- nivelul 2 –funcționarii de vârf, managerii medii –sunt deserviți de sisteme bazate pe spreadsheeturi, având și elemente de inteligență artificială (în general sisteme de asistare a deciziilor).
- nivelul 3 –managerii de vârf, top-managerii –activează într-un mediu informațional puternic nestructurat și se bazează pe sisteme având la bază inteligența artificială. Ei lucrează cu informații cu caracter incert (fuzzy), informații ce se obțin prin prelucrarea unui volum mare de date nestructurate. La acest nivel tehnicile statistice clasice sunt adeseori ineficiente în luarea deciziilor.

4.2.1.2. Rolul sistemului informațional

Sarcina fundamentală a sistemului informațional constă în furnizarea la momentul optim a informațiilor necesare pentru fundamentarea deciziilor precum și a informației necesare pentru coordonarea și reglarea modului de îndeplinire a lor. Considerăm că un sistem informațional bine conceput trebuie să aibă următoarele caracteristici:

- flexibilitate și adaptabilitate
- să fie suplu și extensibil
- să asigure o informare cât mai completă și să asigure un volum optim de informare corespunzător fiecărui nivel ierarhic de conducere. Informația trebuie să fie exactă, să corespundă cantitativ și calitativ, să nu fie redundantă sau incompletă.
- să fie rapid și să aibă un cost de informare scăzut.

Necesitățile informaționale diferă în funcție de piramida ierarhică la care ne referim: la primul nivel (la baza piramidei) se lucrează cu informații structurate; pe măsură ce înaintăm în organigrama întreprinderii, informația tinde să devină mult mai diversificată și mai puțin structurată. Sistemul informațional trebuie să asigure informații de sinteză factorilor decidenți din cadrul firmei pe de o parte, iar de pe altă parte din mediul economic în care firma își desfășoară activitatea. Importanța informațiilor de natură exogenă are un rol critic în bunul mers al unei firme (în cazul dat, politica de comisioane a firmei, nivelul minim al investiției, facilități acordate clienților, etc).

Majoritatea autorilor sunt de acord cu următoarea clasificare a sistemelor informaționale⁴¹:

- Sisteme informaționale de prelucrare a tranzacțiilor (Transaction Processing Systems TPS) - preiau tranzacțiile generate de activitatea firmei în bazele de date interne și constituie infrastructura următoarelor niveluri ale sistemelor informaționale.
- Sisteme informaționale pentru conducere (Management Information Systems MIS) -pleacă de la TPS și sintetizează informațiile sub formă de rapoarte periodice destinate în general nivelurilor intermediare de conducere și au ca finalitate controlul.

⁴¹. [Racovițan96] –Racovițan D, Nițchi S și colectiv–Bazele prelucrării informațiilor și tehnologie informațională, Editura Intelcredo, Deva, 1996.

- Sisteme de asistare a deciziilor (Decision Support Systems DSS) –despre care vom vorbi pe larg în subcapitolul 4.2
- Sisteme informaționale pentru conducerea executivă (Executive Information Systems EIS) –sunt, de fapt, DSS destinate nivelurilor superioare de conducere.

Schematic, legăturile dintre aceste sisteme informaționale pot fi reprezentate astfel (figura 15):

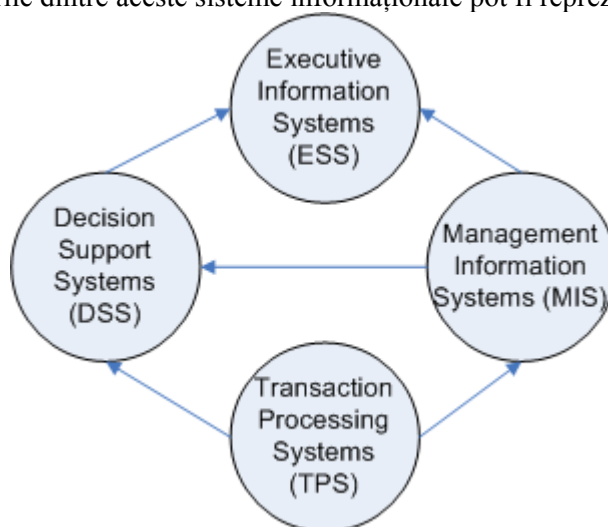


Fig. 15 Legăturile dintre sistemele informaționale

4.2.1.3. Decizia în managementul firmei

Decizia constituie punctul central al activității de management, indiferent de domeniu, deoarece ea se regăsește în toate funcțiile activității de conducere. Întreaga activitate a unei firme este o înlănțuire de decizii fără de care nu ar putea să funcționeze normal. Deciziile trebuie să fie rezultatul unui proces de gândire precedat de o informare și o analiză temeinică a tuturor datelor problemei, a elementelor de influență, având în vedere condițiile concrete în care își desfășoară activitatea fiecare firmă în parte.

Decizia se definește ca o linie de acțiune aleasă în mod conștient de către conducerea întreprinderii în scopul atingerii unui anumit rezultat și reprezintă un act de creație în care se împletesc gândirea logică și factorii psihologici. Decizia rezultă ca urmare a prelucrării unor informații și cunoștințe de către o persoană sau grup de persoane care constituie un organism decizional.

Procesul de luare a unei decizii poate fi reprezentat astfel (figura 16):

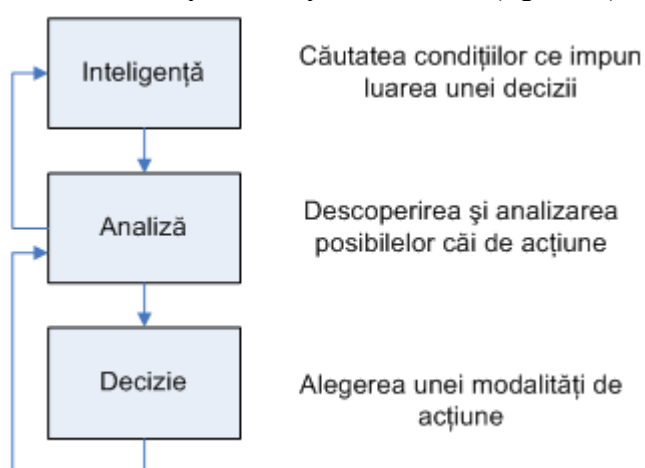


Fig.16 Procesul de luare a unei decizii

Sistemul decizional este reprezentat de ansamblul elementelor interdependente care determină elaborarea și fundamentarea deciziilor. Elementele componente ale sistemului decizional din cadrul firmelor sunt aproximativ aceleași cu caracteristicile specifice domeniului și complexității fiecărei firme.

Dacă analizăm procesul decizional în conexiune cu celelalte procese care au loc în cadrul firmei, se constată existența unor succesiuni de cicluri procesuale cuprinzând:

- a) adoptarea deciziei
- b) aplicarea în practică a deciziei
- c) înregistrarea efectelor

Deciziile manageriale, pentru a conduce la realizarea obiectivelor stabilite, trebuie să răspundă unor cerințe dintre care cele mai importante considerăm că sunt următoarele:

1. *Fundamentarea științifică* a deciziei impune luarea în considerare a condițiilor concrete din firmă, recurgerea la metode adecvate de fundamentare și valorificare a experienței practice a decidenților.
2. *Adoptarea* deciziilor trebuie făcută de către organe sau persoane care au dreptul și împuternicirea de a le lua, pentru a asigura însușirea și răspunderea pentru deciziile luate.
3. *Coordonarea* deciziilor –deciziile trebuie să conducă la realizarea obiectivului de ansamblu al firmei.
4. *Legalitatea* deciziilor -adoptarea deciziilor trebuie făcută doar de către persoane care sunt investite cu drept legal și împuternicite în acest sens.
5. Deciziile trebuie să fie *oportune* (mai bine o decizie bună la timp decât una foarte bună luată cu întârziere), eficiente, complete (să conțină toate elementele necesare înțelegerii și luării în cunoștință de cauză a deciziilor). Pentru aceasta, la formularea deciziei trebuie să se specifice obiectivul urmărit, modalitatea de acțiune, responsabilul cu luarea deciziei, termene de aplicare și executare, fondurile necesare în vederea implementării deciziilor.

4.2.1.4. Tipuri de decizii

Din punctul de vedere al tipului de decizie deosebim:

- *Decizii de rutină* -nu prezintă complexitate pentru un manager deoarece presupun o combinare între o problemă cunoscută și o soluție cunoscută, deja aplicată în rezolvarea unor probleme frecvente și care dă rezultate bune. Efortul de creație al managerului este minim, el respectă toți pașii logici ai fundamentării deciziei doar că se parcurg cu o mare rapiditate.
- *Decizii adaptive* –presupun existența unei doze de incertitudine. Au o importanță deosebită pentru evoluția sistemului deoarece pe măsură ce ele sunt fundamentate și implementate permit adaptarea sistemului la condițiile pieței.
- *Decizii inovative* –sunt cele mai dificile. Implică multă incertitudine fiind legate de un nivel ridicat al riscului. Acest tip de decizii se asimilează, de regulă, cu acțiunile de penetrare pe piețe necunoscute, restructurări, re tehnologizări, operații speculative pe piață. Fundamentarea și adoptarea lor se face într-o perioadă mare de timp, presupun acceptarea riscului și a responsabilitățile pentru efectele ce derivă din acestea. Succesul acestui tip de decizie ține în mare măsură atât de informațiile disponibile cât și de abilitatea managerului de a stabili cu anticipație situațiile posibile și de a prevedea impactul lor potențial asupra evoluției viitoare a întreprinderii.

În funcție de numărul de persoane care participă la adoptarea unei decizii, există decizii fundamentate și elaborate de o singură persoană sau de un grup de persoane. Multe decizii, în special manageriale sau strategice sunt luate de grupuri mai degrabă decât de persoane individuale. DSS, EIS și ES pot oferi suportul unor grupuri dar nu oferă posibilitatea unei decizii de grup; un astfel de suport este oferit de o tehnologie numită Sisteme de asistare a deciziilor de grup (Group Decision Support System – GDSS).

În funcție de periodicitatea elaborării deciziei, avem:

- *Decizii periodice* elaborate la anumite intervale de timp
- *Decizii aleatoare* adoptate la intervale neregulate, greu de anticipat, necesitatea fundamentării lor fiind determinată de factori necontrolabili; eficacitatea lor depinde decisiv de potențialul decizional al decidentului
- *Decizii unice* care reprezintă un caz de excepție, neavând precedent și care probabil nu se vor repeta într-un viitor apropiat; eficacitatea lor depinde decisiv de potențialul decizional al decidentului.

După gradul de structurare, deciziile se împart în:

- *Decizii structurate* –sunt deciziile de rutină, se iau în condiții de certitudine, pentru aceste probleme existând algoritmi preciși de rezolvare
- *Decizii nestructurate* –acestea fac apel la judecata și intuiția decidentului; acestea sunt în general decizii importante, cu un caracter inovator și au consecințe importante în bunul mers al firmei. În general elementele acestui tip de decizii sunt calitative și neprogramabile.
- *Decizii semistructurate* –în cadrul lor există elemente preponderent cantitative dar obiectivele și scopurile nu sunt precise; se pot aplica proceduri cunoscute doar parțial. În general acest tip de probleme se împart în părți structurate și părți nestructurate.

Măsura gradului de structurabilitate al unei probleme se determină nu doar în funcție de situația decizională în cauză ci și în funcție de modul în care este ea percepută de către un anumit decident. Potrivit lui Gorry și Scott-Morton⁴², controlul stocurilor, ordonanțarea producției și planificarea cercetării-dezvoltării de produse și procese sunt exemple de procese de decizie structurate, semistructurate și, respectiv, nestructurate.

După nivelul la care au loc, Nof⁴³ clasifică deciziile astfel:

Nivel	Exemplu	Structură	Caracteristici	Restricții de timp
Strategic	planificare generală	nestructurat semirepetitiv	în condiții de risc	nu
Tactic	gestiunea stocurilor	–"–	parțial ad-hoc parțial distribuit	uneori
Operativ	conducerea atelierului	semistructurat	risc redus ad-hoc distribuit	timp real

În funcție de acest criteriu putem vorbi despre:

- *Decizii strategice* –determină obiectivele, resursele și politicile firmei. Implică în general un număr mic de persoane.
- *Decizii tactice*, de control managerial –sunt legate de urmărirea eficienței cu care sunt utilizate resursele
- *Decizii operaționale* –privesc trasarea de sarcini stabilite la nivelurile superioare.

După gradul de cunoaștere a condițiilor mediului ambiant, există trei tipuri de decizii:

- *În condiții de certitudine* –acestea sunt caracterizate prin manifestarea unei singure stări a condițiilor obiective. Cunoscându-se probabilitatea de apariție a respectivei stări a condițiilor obiective, se cunosc și variantele de realizare a obiectivelor.

⁴² [Gorry71] Gorry, G.A., M.S. Scott-Norton -A framework for management information systems. Sloan Management Review, 1971

⁴³ [Nof81] Nof S.Y. - Theory and practice in decision support for manufacturing control; Data Base Management: Theory and Applications, 1981

- *În condiții de risc* –deciziile în condiții de risc presupun existența mai multor stări ale condițiilor obiective pentru care se cunosc probabilitățile de apariție. Elaborarea unei decizii în condiții de risc trebuie să se bazeze pe modelele matematicii probabilistice.
- *În condiții de incertitudine* –în acest caz, nu se cunosc probabilitățile de apariție a stărilor condițiilor obiective. În acest caz, probabilitatea trebuie apreciată de decident.

În funcție de numărul de criterii care stau la baza deciziei avem:

- *Deciziile unicriteriale*, când se urmărește un singur scop (maximizarea profitului unei investiții)
- *Decizii multicriteriale* (se urmărește pe lângă maximizarea profitului și lichiditatea, riscul precum și alți indicatori).

4.2.1.5. Tratarea incertitudinii

Reacția intuitivă a managerilor este de a lua decizii în cazul incertitudinii numai când se cunosc șansele de apariție ale fiecărui fenomen. În acest caz problema devine una de risc. Totuși, dacă nu sunt informații care să ofere riscul fiecărei alternative, trebuie luate decizii în condiții de incertitudine. De exemplu, o vedere optimistă în privința posibilităților investiționale implică obținerea celui mai mare randament selectând investiția în acțiuni. O vedere pesimistă implică considerarea celui mai proaste alternative posibile și în consecință investiția în certificate de depozit. Astfel de încercări de tratare a incertitudinii au serioase deficiențe. De aceea, considerăm că în vederea luării unei decizii trebuie să colectăm suficiente informații astfel încât problema să poată fi tratată sub un risc asumat.

4.2.1.6. Tratarea riscului

În economia de piață, al cărei mecanism se bazează pe concurență, pe oscilația prețurilor, incertitudinea și riscul sunt niște variabile de care trebuie ținut seama, suprimarea lor nefiind posibilă. Perceperea riscului, conturarea dimensiunilor sale, presupune o conștientizare a pericolului investițiilor în acțiuni, asumarea unor responsabilități de către o persoană sau un grup de persoane. Oricâte informații ar fi disponibile, decidenții nu vor putea niciodată să elimine incertitudinea și riscul din activitatea lor; tot ceea ce pot face este să-l studieze și să-l cuantifice, pentru a-l cunoaște și a alege varianta cea mai puțin riscantă. Condițiile de risc sunt inerente procesului de conducere și apar și ca urmare a faptului că nu sunt obținute în timp util și valorificate informații care să-i ofere decidentului un control deplin asupra fenomenelor ce se vor produce în viitor. Considerăm că pentru a atenua riscul chiar și atunci când informația nu este suficientă calitativ și cantitativ, managerii trebuie să fie preocupați de a identifica cauzele potențiale și probabilitatea cu care poate să apară situația preconizată.

Presupunem o companie care poate folosi următoarele metode de plasament: obligațiuni, acțiuni și certificate de depozit. Compania poate fi interesată doar de maximizarea venitului adus de respectiva investiție sau urmărește și alte scopuri precum lichiditatea și riscul investiției. În acest din urmă caz problema va fi una multicriterială. Venitul poate depinde de starea economiei care poate fi de creștere pronunțată, stagnare sau recesiune. Considerăm un exemplu în care experții estimează următoarele valori ale rentabilităților investițiilor:

1. În cazul unei creșteri economice, obligațiunile vor aduce o rentabilitate de 12%, acțiunile 15%, iar certificatele de depozit 6,5%.
2. În cazul stagnării, aceste procente sunt previzionate astfel: obligațiuni 6%, acțiuni 3% și certificate de depozit 6,5%.
3. În cazul recesiunii, obligațiunile vor avea o rentabilitate de 3%, acțiunile vor scădea la 2%, iar certificatele de depozit vor aduce o rentabilitate de 6,5%.

Problema este de a selecta cea mai bună alternativă investițională. Aceasta se poate realiza pe baza unui model matematic care include: variabile de decizie (alternative), variabile incontroabile (starea economiei) și variabile rezultat. Se disting două probleme: incertitudinea și riscul. În cazul incertitudinii nu cunoaștem probabilitățile fiecărei stări ale naturii pe când în cazul riscului se cunosc probabilitățile de apariție ale fiecărei stări ale naturii.

Presupunem că probabilitatea de creștere economică este de 50%, de stagnare 30%, iar de inflație 20%. În acest caz, tabela de decizie arată astfel⁴⁴:

Alternativa	Creștere economică 0,50	Stagnare 0,30	Inflație 0,20	Randament așteptat
Obligațiuni	12,0%	6,0%	3,0%	8,4%
Acțiuni	15,0%	3,0%	2,0%	8,0%
Certificate de depozit	6,5%	6,5%	6,5%	6,5%

Cea mai comodă metodă pentru rezolvarea unei astfel de probleme este selectarea alternativei cu cel mai bun rezultat. Rezultatul se obține prin înmulțirea randamentelor cu probabilitatea aferentă și însumarea lor. De exemplu, în cazul obligațiunilor avem:

$$12 \cdot 0,5 + 6 \cdot 0,3 + 3 \cdot 0,2 = 8,4.$$

Deci investind în obligațiuni vom avea o rentabilitate medie așteptată de 8,4%.

În cazul în care urmărim mai multe scopuri, cum ar fi rentabilitatea, siguranța și lichiditatea avem de-a face cu o decizie multicriterială. În acest caz, unele date pot să nu fie numerice ci simbolice, așa cum rezultă din tabelul următor:

Alternativa	Rentabilitate	Siguranță	Lichiditate
Obligațiuni	8,4%	Ridicată	Ridicată
Acțiuni	8,0%	Scăzută	Ridicată
Certificate de depozit	6,5%	Foarte ridicată	Ridicată

Riscul unei acțiuni, concretizat într-o valoare mică sau nulă a dividendelor și instabilitatea cursului, este dat însăși de riscul activității întregii firme. Posibilitatea oferirii către acționari a unui câștig substanțial poate rezulta doar în urma unei activități rentabile care să asigure o mărime a profitului suficient de mare atât pentru plata acționarilor cât și pentru buna desfășurare a activității. Evaluarea bursieră a capitalului unei întreprinderi prin intermediul cotației acțiunilor sale este de fapt rezultatul unei evoluții pe piață precum și a percepției investitorilor asupra acestui lucru. Cursul stabilit prin libera confruntare a cererii cu oferta pe piața mobilă este un efect al unei cauze mult mai profunde: performanțele economice ale firmei.

Autorul consideră că evaluarea corectă a raportului dintre risc și câștig poate induce oportunități extraordinare pentru cei care știu să gestioneze acest risc. Scopul unui investitor este nu evitarea completă a riscului ci evitarea riscurilor puțin înțelese. Evitarea riscurilor poate fi o cale spre eșec. Istoria ne-a arătat că marile împliniri au presupus întotdeauna asumarea unor riscuri substanțiale într-o formă sau alta.

⁴⁴ [Zaharie01] Zaharie D., Albescu F. și alții –Sisteme informatice pentru asistarea deciziei, Editura Dual Tech, 2001

4.2.1.7. Etapele procesului decizional

Luarea unei decizii este necesară în anumite momente de timp, atunci când apar situațiile decizionale. Apariția stimulilor care creează condițiile ce determină necesitatea alegerii dintre mai multe alternative în vederea atingerii unei stări dezirabile conturează o problemă decizională. *Procesul decizional* este constituit dintr-o serie de activități decizionale.

Conform lui Simon⁴⁵, etapele *procesului decizional* sunt:

1. *Identificarea și definirea problemei*, sesizarea necesității luării deciziei. Rolul decidentului constă la început în identificarea problemei decizionale și diferențierea ei de problemele nedecizionale. După identificarea problemei, este necesară definirea ei și prezentarea principalelor sale caracteristici. Necunoașterea în întregime a problemei poate compromite procesul decizional chiar dacă celelalte etape sunt parcurse corect. Cunoscutul specialist american în domeniul managementului, P. Drucker a afirmat că “un răspuns corect la o problemă greșit pusă este lucrul cel mai inutil dacă nu chiar periculos”. Un sistem expert poate ajuta în proiectarea fluxului de informații și în interpretarea informațiilor colectate. Deoarece multe informații sunt incerte se pot folosi combinații de sisteme expert și rețele neuronale.
2. *Analiza*. Odată problema identificată, trebuie să se treacă la etapa analizei. Analiza poate fi cantitativă, calitativă sau combinată. Analiza cantitativă poate fi făcută de un DSS iar cea calitativă de un ES. Se stabilesc criterii și obiective decizionale. Criteriile decizionale sunt puncte de vedere ale decidentului cu ajutorul cărora sunt izolate anumite aspecte ale realității socio-economice. Criteriile decizionale se caracterizează prin mai multe niveluri de reprezentare, fiecare reprezentând obiective posibil de realizat. Obiectivul unui proces decizional este nivelul propus pentru a fi realizat din respectivul criteriu. De exemplu, dacă criteriul de decizie este “profitul”, obiectivul este “maximizarea profitului”.
3. *Luarea unei decizii bazată pe rezultatele analizei*. Acesta este una din cele mai importante acțiuni și stabilește care dintre variantele decizionale posibile corespunde variantei optime conform criteriilor decizionale utilizate. Ea poate fi făcută având la bază un DSS.
4. *Implementarea* -În cazul în care decizia este de implementare a soluției propuse, acesta constă în aplicarea variantei optime. Un rol important îi revine decidentului în ceea ce privește motivarea și transmiterea deciziei luate. În cadrul acestei etape un DSS sau un ES poate oferi suportul.

Aceste etape sunt reprezentate în figura 17.

⁴⁵ [Simon60] Simon, H –The New Science of Management Decision, New-York, Harper & Row, 1960.

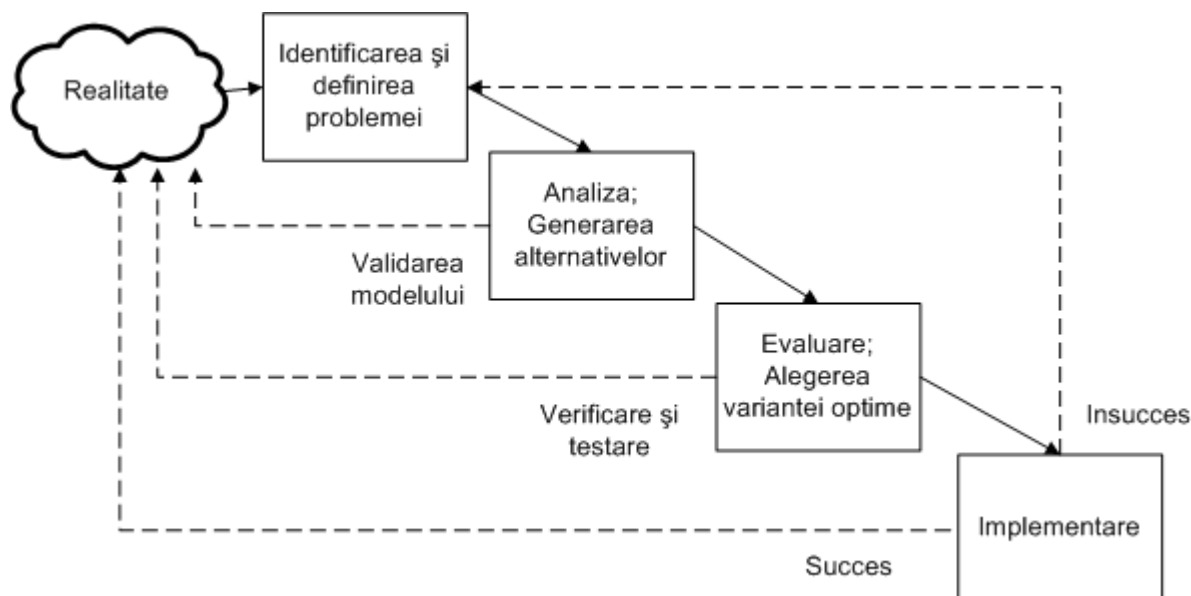


Fig. 17 Etapele procesului decizional

Filip⁴⁶ afirmă că *procesul decizional* este constituit dintr-o serie de activități decizionale. El începe cu conștientizarea unei situații decizionale și culegerea datelor, continuă cu proiectarea alternativelor și a modelelor, alegerea unei soluții și se încheie cu adoptarea unei decizii, lansarea ei spre execuție și evaluarea rezultatelor acțiunii.

4.2.1.8. Principalele categorii de instrumente informatice folosite în procesul decizional

O dată cu trecerea timpului, asistarea informatică a activității manageriale și-a manifestat tot mai evident capacitatea de a contribui decisiv, alături de profesionalismul managerilor la asigurarea calității deciziilor și implicit la creșterea performanțelor organizațiilor. Afirmarea informaticii manageriale ca o direcție nouă în informatică s-a concretizat în crearea unor instrumente specifice asistării decizionale. Sheridan⁴⁷ susține că această asistare poate îmbrăca diverse forme:

- furnizarea de informație de intrare în procesul decizional; în acest caz, decizia este procedural independentă de sistemul informatic dar calitatea ei este influențată de cea a informației de intrare
- analize de tipul “What If?” (“Ce se întâmplă dacă?”) a alternativelor decizionale propuse de către decident
- generarea unor soluții decizionale care pot fi modificate, validate și însușite de către factorii decizionali implicați
- generarea unor soluții decizionale normative, ce nu pot fi amendate, ci doar admise sau respinse

Măsura în care soluționarea problemelor manageriale poate beneficia de facilitățile asistării informatice depinde de gradul de structurare a respectivelor probleme⁴⁸. Dacă o problemă ar fi complet structurată, un instrument automat ar putea-o rezolva fără nici o intervenție umană. În schimb, dacă o problemă nu prezintă nici un fel de structură, nu se poate conta decât pe ajutorul hazardului.

Managerii reprezintă o categorie de beneficiari ai asistării informatice nu numai extrem de influentă –în general, dar și în particular, față de soluția de informatizare propusă- ci și exigentă; pentru ei sunt considerate atractive sistemele informatice care îi pot ajuta să identifice și să înțeleagă probleme de

⁴⁶ [Filip04] Filip F.G. –Sisteme suport pentru decizii, Editura Tehnică, București, 2004.

⁴⁷ [Sheridon80] Sheridan A. –Decision Support Systems. Current Practices and Continuing Challenges. Addison Wesley, Reading, Massachusetts, 1980.

⁴⁸ [Filip00] Filip F.G., Filip C.I. –Decision Support Systems: from Job Aids to Intelligent Assistants. Preprints MIM 2000, IFAC Symposium on Manufacturing, Modelling and Control, July 12-14, Patras, 2000.

decizie și apoi să le rezolve, oferindu-le în timp real și în mod selectiv, informație validă și diversă, preponderent sintetică, provenită din surse eterogene intra și extra-organizaționale⁴⁹.

Caracteristicile diferitelor categorii de instrumente informatice folosite în procesul decizional pot fi redate comparativ astfel⁵⁰:

<i>Categoriile de instrumente informatice</i>	<i>Obiectul asistării decizionale</i>	<i>Utilizatori</i>	<i>Rezultate furnizate</i>	<i>Operații efectuate</i>	<i>Orientarea în raport cu timpul</i>
Sisteme informatice tradiționale	Prelucrări de date	Decidenți de la toate nivelurile	Rapoarte	Calculare numerice, editări și regăsiri de date	Retrospectivă
Sisteme de asistare a deciziilor	Asistarea deciziei în probleme semistructurate	Funcționari și decidenți	Alternative decizionale și evaluări ale deciziilor umane	Analize “on line”, simulări, optimizări, asistarea comunicării	În timp real sau prospectivă
Sisteme cu inteligență artificială	Asistare în probleme slab structurate	Profesioniști ce dețin sau nu cunoștințe de tip expert	Recomandări, diagnoze, planuri, proiecte	Inferențe	Predominant prospectivă

Aplicațiile pe calculator ce servesc procesului decizional sunt în creștere pronunțată. Calculatoarele sunt prezente din ce în ce mai des pe birourile managerilor care pot accesa mii de baze de date de pe întreg cuprinsul mapamondului. Tot mai multe firme nu mai iau decizii fără a folosi analize având la bază aplicații bazate pe calculator. Costul echipamentelor și a programelor este într-o continuă scădere în timp ce performanțele sistemelor informatice sunt în continuă creștere. Managerii pot lua decizii mai bune deoarece dispun de informații mai precise, la îndemână sub o formă din ce în ce mai accesibilă.

Sistemele de asistare a deciziilor (Decision Support System – DSS, numite și Sisteme interactive de asistare a deciziilor SIAD) împreună cu Sistemele de asistare ale executivului (Executive Information System –EIS) și Sistemele expert (Expert System –ES) formează Sistemul de asistare al managementului (Management Support System –MSS) și au un impact din ce în ce mai mare în procesul decizional managerial.

Contribuția unui sistem de asistare a deciziilor, privită din perspectiva a ceea ce oferă pe de o parte factorul uman, iar pe de altă parte din perspectiva generală a domeniului tehnologiei informației poate fi schematizată astfel:

<i>Factorul uman</i>	<i>DSS</i>	<i>Tehnologia informației</i>
Experiență	Creșterea productivității	Viteză
Intuiție	Creșterea gradului de înțelegere	Informații
Judecată	Creșterea flexibilității	Capacitate mare de procesare
Cunoaștere	Scăderea complexității problemelor	
	Scăderea costurilor	

⁴⁹ [Dragomirescu96] Dragomirescu H. –Strategii de informatizare –în Strategii manageriale de firmă (coordonator Nicolescu O.), Editura Economică, București, 1996.

⁵⁰ [Millet92] Millet I., Mawhinney C. –Executive Information Systems, Information & Management, 1992.

4.2.2. Conceptul de sistem de asistare a deciziilor

4.2.2.1. Noțiuni introductive

Primul care a propus termenul de “Decision Support System” a fost Michael Scott Morton în februarie 1964. De atunci, conceptul a fost definit în diverse moduri de către diverși autori.

În 1970, Little a definit *sistemele de asistare a deciziilor* ca fiind: “seturi de proceduri de procesare a datelor și modelelor în scopul asistării managerilor în procesul de adoptare a deciziilor”. În 1971, M. Scott-Morton a definit sistemele informatice de asistare a deciziilor (SIAD) ca fiind “sisteme informatice interactive, care ajută decidentul în utilizarea datelor și modelelor în scopul rezolvării problemelor nestructurate și semistructurate”. Keen și Morton (1978) definesc un SIAD astfel⁵¹: “SIAD combină resursele intelectuale ale decidenților cu capacitățile calculatorului în scopul îmbunătățirii calității procesului decizional. El reprezintă un sistem informatic destinat managerilor care adoptă decizii pentru probleme semistructurate”. La nivelul anilor '70, primele sisteme de asistare a deciziilor se fondau pe probleme clasice, adică pe decizii structurate. În aceste cazuri, formalizarea era deja realizată. Acestea includeau statistici ale cercetării operaționale și aveau la bază algoritmi de optimizare și calcule numerice sau erau îndreptate exclusiv spre gestiunea informației (baze de date, gestiunea fișierelor, fluxuri de informații în întreprindere).

În a doua jumătate a anilor 1980, se renunță la restricționarea domeniului de aplicație la problemele semistructurate și nestructurate. Conceptul de structurare a unei probleme a început să fie considerat ca irelevant. Moore și Chang definesc în 1986 SIAD-urile ca fiind sisteme flexibile, capabile să asiste analiza datelor și modelarea deciziilor, utilizate în regim discontinuu, atât pentru probleme de rutină cât și pentru situații complet noi. Tot în această perioadă se conturează ideea asistării rezolvării problemelor nestructurate cu tehnologii inteligente. Keen (1980) utilizează termenul SIAD pentru acele situații în care un sistem “final” nu poate fi realizat decât în urma unui proces flexibil de învățare și evoluție. În această perioadă, sistemele de asistare a deciziilor au fost construite în jurul unor baze de modele. Aplicațiile realizate lăsa la latitudinea utilizatorilor posibilitatea variației modelelor reale în cadrul unei structuri fixate. Tipul de model este fixat iar caracteristicile sale variază într-un câmp restrâns, dând naștere la sisteme de asistare pentru decizii unistructurale. Dacă sunt legate între ele modele cu structuri foarte diferite, atunci vorbim de sisteme de asistare pluristructurale.

DSS, ca sisteme informatice de asistare a deciziilor, au parcurs până astăzi patru stadii de evoluție: sistemele decizionale, sistemele de asistare a deciziilor cu explorare euristică, DSS inteligente (integrarea cu bazele de cunoștințe) și tehnologiile noi ale informaticii decizionale cum ar fi: Data Mining, Data Warehousing, baze de date multidimensionale și tehnologiile OLAP.

*Sistemul informatic de asistare a deciziei*⁵² se constituie într-un sistem care este utilizat în mod direct de decident (oferindu-i informații privind cauzele și căi de acțiune posibile pentru rezolvarea unei anumite probleme). Acest tip de sistem informatic permite utilizatorului să controleze procesul decizional, oferind modele de rezolvare, metode de analiză și agregare a informațiilor, fără însă a impune proceduri sau soluții predefinite.

⁵¹ [Morton71] Morton S –Management Decision Systems: computer based support for decision making; Harvard University Press, Cambridge, 1971.

⁵² [Zaharie01] Zaharie D., Albescu F. și alții –Sisteme informatice pentru asistarea deciziei, Editura Dual Tech, 2001

Scott Morton indică patru caracteristici ale acestor sisteme:

1. DSS încorporează atât date cât și modele
2. Sunt destinate să asiste managerii în procesul decizional în probleme în general semistructurate
3. Ele ajută, nu înlocuiesc procesul decizional
4. Ele urmăresc mai mult îndeplinirea anumitor scopuri decât eficiența rezultatului.
Alte caracteristici ale sistemelor de asistare a deciziilor sunt⁵³:
 - interactivitate la nivel înalt
 - abilitatea de a lucra cu probleme complexe
 - căutarea soluțiilor necesită manipulări de date, căutare de informații de tip euristic
 - procedurile care urmează să se execute la un moment dat nu sunt cunoscute apriori, acestea depinzând de date sau de rezultatele intermediare
 - criteriile de decizie sunt numeroase, conflictuale și depind adesea de utilizatori
 - răspunsuri rapide la situații neașteptate datorită schimbării unor condiții
 - posibilitatea de a urmări mai multe strategii pentru diferite configurații
 - reduceri de costuri (mai ales costurile unor decizii greșite)
 - experiența, intuiția, judecățile și preferințele decidentului sunt esențiale.
 - posibilitatea de a fi accesate atât individual cât și de mai multe persoane.
 - în general pot fi adresate ad-hoc, în probleme deosebite, dând o reprezentare validă a unei lumi reale complexe prin modele ce pot sta la baza unor previziuni necesare în astfel de situații.

Principalele operații realizabile într-un sistem de asistare a deciziilor sunt: gruparea și agregarea datelor; clasificarea, organizarea și ordonarea datelor; calcule; realizarea unor statistici; generarea scenariilor; realizarea previziunilor și evaluarea; rezumarea, analiza, interpretarea unui ansamblu de date; crearea de rapoarte și grafice. Mai mulți autori consideră că există o sinergie naturală între tehnicile de structurare a problemelor utilizate în baza de modele a unui sistem de asistare a deciziilor și arhitecturile bazate pe reguli din cadrul unui sistem expert. Astfel, tehnicile de modelare ale bazei de modele sunt potrivite pentru o structurare inițială a problemei, iar sistemele expert pot folosi această structură dezvoltând-o în continuare și asigurând o interfață familiară cu utilizatorul. Sistemele expert pot fi utilizate ca o interfață inteligentă între utilizator și un set de modele cantitative gestionate de o aplicație de asistare a deciziilor, dar mai ales pentru a asigura analiza aspectelor calitative, nestructurate din cadrul domeniului respectiv.

Una din particularitățile acestor sisteme este integrarea lor pe scară tot mai largă, conținând adesea mai multe instrumente elementare interconectate. Există diverse posibilități de integrare a sistemelor expert în sistemele de asistare a deciziei⁵⁴: sisteme expert atașate componentelor SIAD, sistemul expert ca o componentă separată a SIAD, sistemul expert ca generator de soluții alternative pentru SIAD, sistem expert și SIAD într-o structură unică. Aceste integrare permite trecerea cu ușurință a datelor dintr-un instrument în altul, de exemplu dintr-un spreadsheet la un software grafic. Această noțiune de multi-instrument facilitează utilizarea sistemelor de asistare a deciziilor și a sistemelor expert. Se poate, de exemplu, interfața un spreadsheet cu un sistem expert pentru a putea răspunde la o problemă care necesită calcule auxiliare. Integrarea mai multor baze de cunoștințe a dus la tehnologia sistemelor multi-expert.

Apariția sistemelor expert pe scena informaticii a determinat și apariția unei noi generații de sisteme suport și anume cea a *sistemelor de asistare pentru decizii inteligente*. Acestea sunt în general pluristructurale, conținând o structurare tip bază de cunoștințe a sistemelor suport pentru decizii inteligente. Acesta este domeniul cel mai nou și mai promițător al sistemelor de asistare a deciziei, obținându-se avantaje în toate fazele procesului decizional.

4.2.2.2. Caracteristicile sistemelor de asistare a deciziilor

⁵³ [Zaharie01] Zaharie D., Albescu F. și alții –Sisteme informatice pentru asistarea deciziei, Editura Dual Tech, 2001

⁵⁴ [Bojan01] Bojan I. B.-Sisteme expert pentru evaluarea întreprinderilor, Editura Dual Tech, 2001.

Cu toate că s-au scurs câteva decenii de la apariția conceptului de sistem de asistare a deciziilor și nu există încă o definiție unanim acceptată, putem prezenta în continuare un subset de caracteristici esențiale care aparent nu sunt contestate de nici un autor^{55 56 57 58}.

1. Oferă suport de decizie îndeosebi în situații semistructurate și nestructurate combinând gândirea umană cu suportul oferit de calculator.
2. Aceste sisteme nu au menirea să se substituie decidentului final ci doar sprijină (în limba engleză *support*) activitățile de elaborare a deciziilor. Cel ce ia decizia are controlul complet asupra tuturor pașilor procesului decizional în rezolvarea problemei. Un DSS oferă un suport și nu înlocuiește decidentul; acesta din urmă poate oricând să treacă peste recomandările oferite.
3. Suportul este oferit pentru diverse nivele de decizie ale managementului de la top-manageri până la șefii de compartimente.
4. Vizează îmbunătățirea eficacității procesului decizional, cu accent pe creșterea productivității muncii decidentului și pe îmbunătățirea deciziilor (calitate, oportunitate, aplicabilitate). Astfel aceste sisteme îmbunătățesc calitatea deciziilor, le măresc acuratețea și ajută luarea lor la timp.
5. Suportul este oferit atât individual cât și grupurilor. Cu cât problemele sunt mai puțin structurate cu atât implică mai mult decizii ale mai multor persoane din diverse compartimente și nivele organizaționale.
6. Oferă suport atât mai multor decizii independente cât și interdependente.
7. Comportă toate fazele procesului decizional: inteligență, proiectare, alegere și implementare.
8. Se bazează pe modele, iar experimentele la care apelează pot conduce la un spor de cunoștințe prin procesul de învățare.
9. Oferă o varietate de procese și stiluri de luare a deciziilor.
10. Se pot adapta în orice moment la schimbări survenite în mediul economic, dând răspunsuri la situații neașteptate. Această facilitate dă posibilitatea unor analize ad-hoc, la timp și rapide.
11. Sunt ușor de folosit, au o interfață prietenoasă, în general au facilități grafice deosebite.
12. Astfel de sisteme sunt în general ușor de construit. În cazul unor sisteme relativ simple, utilizatorii finali pot fi capabili să-și construiască astfel de sisteme. În cazul sistemelor de dimensiuni mari, acestea se realizează în cadrul unor firme specializate ce dispun de specialiști în sisteme informatice.

Avantajele majore ale utilizării DSS pot fi sintetizate astfel⁵⁹:

- Abilitatea de a da soluții în probleme decizionale complexe
- Răspuns rapid în cazul unor situații neprevăzute datorate schimbării condițiilor de mediu
- Abilitatea de a testa diverse strategii în diferite contexte
- Eficiență și învățare printr-un studiu al sensibilității modelului în analiza de tip “what-if” fapt benefic pentru utilizatorii mai puțin experimentați
- Comunicare facilă între manageri
- Îmbunătățirea controlului și a performanțelor manageriale
- Reducerea costurilor prin eliminarea unor decizii eronate
- Creșterea consistenței deciziilor în raport cu deciziile bazate pe intuiție
- Creșterea eficacității activității manageriale

⁵⁵ [Keen76] Keen P.G. –Interactive computer systems for managers: a modest proposal. Sloan Management Review, Fall, 1976

⁵⁶ [Sprague82] Sprague R.H., Carlson E.D. –Building Effective Decision Support Systems, Prentice Hall, New Jersey, 1982.

⁵⁷ [Turban01] Turban E. Aronson J.E. –Decision Support Systems and Intelligent Systems, Prentice Hall, 2001.

⁵⁸ [Marakas03] Marakas G.M. –Decision Support Systems and Megaputer, Prentice Hall, New Jersey, 2003.

⁵⁹ [Bădescu98] Bădescu A. –Inteligență artificială și sisteme expert, Editura Era, București, 1998.

4.2.2.3. Structura unui sistem de asistare a deciziilor

Un DSS este compus în general din⁶⁰:

1. Componenta de gestiune a datelor -este alcătuită din baza de date a DSS, sistemul de gestiune a bazei de date (Data Base Management System -DBMS), dicționarul bazei de date, facilități de interogare a datelor. Această componentă include informații relevante pentru problema dată, date manipulate de un sistem de gestiune a bazelor de date.
2. Componenta de gestiune a modelelor -include o colecție de modele financiare, statistice, de management sau alte modele cantitative ce conferă facilități de analiză sistemului.
3. Componenta de gestiune a dialogului -constituie interfața utilizatorului cu sistemul și asigură comunicarea între calculator și utilizator.
4. Componenta de gestiune a cunoștințelor este o componentă opțională ce poate acționa ca un subsistem independent sau poate interacționa cu celelalte componente ale DSS. Această componentă apare în cazul integrării sistemului de asistare a deciziei cu un sistem expert, în vederea obținerii unui sistem de asistare a deciziilor inteligent (expert). Modulul cuprinde, în general, motorul de inferență și baza de cunoștințe.

Cele patru elemente constituie arhitectura soft a unui DSS. Modelul conceptual al unui DSS poate fi redat conform figurii nr. 18:

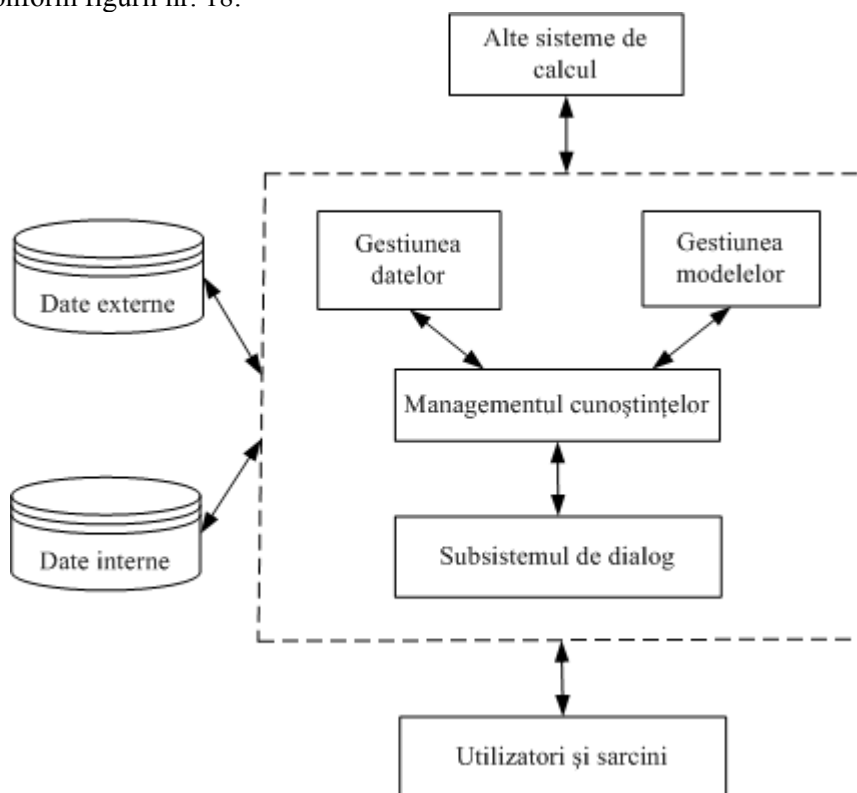


Fig. 18 Modelul conceptual al unui sistem de asistare a deciziilor

4.2.2.4. Fazele dezvoltării unui sistem de asistare a deciziilor

Dezvoltarea unui sistem informatic reprezintă un eveniment cu consecințe majore pentru organizația în cauză. În general acesta urmează același ciclu de viață ca orice produs informatic. Dezvoltarea unui DSS implică următoarele etape:

1. *Planificarea* -punctul de plecare reprezintă diagnosticul problemei; în această etapă se definesc obiectivele și scopurile. Un pas crucial este determinarea deciziilor cheie -de exemplu în cazul unui sistem de management al portofoliului, decizia cheie este determinarea precisă a

⁶⁰ [Bădescu98] Bădescu A. -Inteligență artificială și sisteme expert, Editura Era, București, 1998.

portofoliului în concordanță cu cerințele clientului. Luarea unei decizii în acest domeniu este dificilă datorită mulțimii de factori implicați –unii clienți sunt conservatori și doresc acțiuni cu risc scăzut, alții, dimpotrivă, preferă acțiuni foarte speculative datorită unei rentabilități potențiale mai ridicate.

2. *Identificarea* unor elemente ce se apropie de nevoile utilizatorului (resurse hard și soft, studii și cercetări pe tema dată făcute în alte organizații etc).
3. *Analiza* –aceasta se constituie dintr-o proiectare conceptuală urmată de un studiu de fezabilitate prin care se încearcă definirea unui model ideal ce poate furniza informații pentru deciziile cheie.
4. *Proiectarea* –include determinarea componentelor, structurii și caracteristicilor sistemului.
5. *Elaborarea* sistemului –poate fi făcută în moduri diferite în funcție de filozofia de proiectare și de uneltele ce vor fi folosite. Elaborarea este de fapt imlementarea tehnică a modelului.
6. *Implementarea* –implică testarea (dacă îndeplinește specificațiile de proiectare), evaluarea (dacă îndeplinește cerințele utilizatorilor), demonstrarea capacității sistemului de a face față nevoilor, familiarizarea tuturor utilizatorilor cu sistemul, punerea în funcțiune.
7. *Întreținerea și adaptarea* sistemului la schimbarea mediului. Acest din urmă proces implică reluarea etapelor anterioare.

Menționăm că toți acești pași sau etape nu sunt secvențiali ci pot exista oricând cicluri și salturi.

4.2.3. Concluzii

Având în vedere problemele abordate în cadrul sistemelor de asistare a deciziilor, *autorul desprinde următoarele concluzii:*

- Sistemele de asistare a deciziilor s-au extins pentru a furniza informații unei categorii din ce în ce mai mare de decidenți, lărgind categoria utilizatorilor acestor instrumente. În consecință procesul de luare a deciziilor a cunoscut transformări substanțiale sub influența dezvoltării rapide a tehnologiei informațiilor.
- Rezultatele a numeroase activități de cercetare-dezvoltare și aplicative au demonstrat că aplicațiile de asistare a deciziilor răspund unor nevoi reale și stringente și că există o piață foarte bine dezvoltată pentru aceste sisteme.
- Experiența a demonstrat că, în numeroase cazuri, problemele manageriale sunt fie prea complexe pentru o formalizare matematică riguroasă fie ar fi prea costisitor ca ele să fie rezolvate utilizând numai tehnici de optimizare și simulare. Pentru a depăși aceste dificultăți se utilizează mai multe alternative la soluțiile bazate pe inteligența artificială, utilizând raționamente similare celor umane⁶¹.

S-a observat că anumite sisteme informatice de asistare a deciziilor sunt orientate către procesele intelectuale din emisfera cerebrală stângă, iar altele către cele din emisfera dreaptă. Pentru emisfera stângă sunt caracteristice: gândirea analitică, sub aspectele ei cantitative și calculatorii, iar pentru cea dreaptă gândirea sintetică, recunoașterea formelor și raționamentele bazate pe analogie⁶². În continuare vom vorbi despre sistemele expert ca aplicații ale inteligenței artificiale, concretizate în elaborarea și utilizarea de sisteme bazate pe cunoștințe, care imită în general procesele de raționament din emisfera stângă a creierului.

4.3. SISTEME EXPERT

4.3.1. Privire generală asupra sistemelor expert

4.3.1.1. Scurtă introducere în evoluția sistemelor expert

⁶¹ [Filip01] Filip F.G. –Sisteme de asistare inteligentă a activității manageriale, în “Sistemul informațional managerial al organizației” –coordonator Nicolescu O., Editura Economică, București, 2001.

⁶² [Mintzberg90] Mintzberg H. –Planning on the Left Side and Managing on the Right. In Creative Management (ed. J. Henry) SAGE Publications, London, 1990.

Sistemele expert (SE) sunt sisteme de programe bazate pe tehnicile Inteligenței Artificiale, care înmagazinează cunoștințele experților umani dintr-un domeniu bine definit și apoi le folosesc pentru rezolvarea problemelor din acest domeniu⁶³. În orice domeniu de activitate există probleme cu un grad ridicat de dificultate, care pot fi rezolvate numai de către experți umani, formați ca specialiști în urma unei vaste experiențe în domeniul respectiv. Un expert uman se formează greu și necesită, pe lângă pregătirea și experiența personală, și calități native. De aceea sisteme expert încearcă să imite în principal raționamentele expertului uman prin raționamente artificiale.

E. Feigenbaum, pionierul inteligenței artificiale, a definit sistemul expert ca pe un “program inteligent pentru calculatorul electronic, care utilizează cunoașterea și proceduri de inferență pentru soluționarea problemelor, care sunt suficient de dificile pentru a necesita expertiză umană pentru soluționarea lor”.

Conform lui Luis E. Frenzel, “un sistem expert este un program particular care încorporează o bază de cunoștințe și un motor de inferențe. Programul se comportă ca un consilier inteligent într-un domeniu particular”.

Definiția profesorilor J. Giarratano și G. Riley (NASA) este următoarea “un sistem expert este un sistem care emulează abilitatea de a lua decizii a expertului uman. Termenul “emulează” înseamnă că sistemul este menit să acționeze în toate privințele ca expertul uman. Emularea este ceva mai mult decât simularea, care cere doar să se acționeze prin imitarea condițiilor realizate”.

Sistemele expert sunt o subclasă a *sistemelor bazate pe cunoștințe* (KBS *Knowledge Based Systems*). Există cazuri în care cunoașterea introdusă într-un astfel de sistem nu provine de la un expert, astfel încât sistemul nu este un sistem expert. Unii autori asociază termenul de sistem expert cu sistemele în care baza de cunoștințe este formată din reguli de producție. Distincția între un sistem bazat pe cunoștințe și un sistem expert a devenit confuză datorită poziției de marketing a furnizorilor de tehnologie. Când AION și AI Corp (ambii furnizori de medii de dezvoltare a sistemelor bazate pe cunoștințe) au fuzionat, rezultând compania Trinzic, niciuna nu a vorbit de sisteme expert sau de sisteme bazate pe cunoștințe. În documentația unui produs KBS de marcă (Level 5 Object), se pomenește rar termenul de sistem bazat pe cunoștințe și deloc termenul de sistem expert. Level 5 Object este promovat ca un mediu de dezvoltare Windows “inteligent” oferind SQL, facilități hyper-media și suport grafic în complectarea facilităților de programare orientată obiect și bazată pe reguli. Un produs european cum este Crystal a fost promovat inițial ca un “mediu de dezvoltare a sistemelor expert” dar mai târziu a fost repositionat ca un limbaj de generația a 4-a bazat pe reguli. Având în vedere că în continuare ne vom referi la sistemele ce pot realiza performanțele de gândire și intuiție similare experților umani, vom folosi termenul de sisteme expert. În figura 19 încercăm să reprezentăm relația sisteme de inteligență artificială – sisteme bazate pe cunoștințe – sisteme expert⁶⁴.

Un sistem expert este similar unui sistem de asistare a deciziilor datorită faptului că ambele prevăd pentru utilizatori un sprijin de înalt nivel la rezolvarea problemelor. Diferențele dintre cele două tipuri de sisteme sunt următoarele⁶⁵:

- un sistem de asistare a deciziilor constă din rutine care reflectă modul în care managerul înțelege să rezolve o anumită problemă precum și stilul și capacitatea profesională a acelui manager. Un sistem expert oferă oportunitatea de a lua decizii ce depășesc capabilitățile managerului. De exemplu, un specialist în proiecte de investiții dintr-o bancă poate folosi un sistem expert proiectat de un expert financiar de clasă mai înaltă, profitând astfel de experiența și cunoștințele superioare integrate în acest sistem expert.
- abilitatea sistemului expert de a explica modul de raționament în atingerea unei anumite soluții particulare. Experiența a arătat că explicarea modului în care s-a obținut o anumită soluție este mai valoroasă decât soluția însăși.

⁶³ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

⁶⁴ [Andone94] Andone I. –Sisteme expert. Principii și dezvoltarea aplicațiilor de gestiune, vol. I, Editura Polirom, Iași, 1994.

⁶⁵ [Coroescu02] Coroescu T –Sisteme informatice pentru management, Editura Lumina Lex, București, 2002



Fig. 19 Relația dintre sistemele de inteligență artificială–sistemele bazate pe cunoștințe–sistemele expert

Autorul consideră că cele mai bune sisteme expert sunt caracterizate prin manevrabilitate, utilizare ușoară și performanțe foarte ridicate. În asemenea sisteme expert sunt incluse baze de cunoștințe speciale cu care se pot rezolva în primul rând probleme neformulate numeric. Deși aceste sisteme nu pot concura întotdeauna cu adevărații experți umani, faptul că accesul la ele este ușor, utilizarea lor este facilă și sunt relativ ieftine, reprezintă o provocare serioasă la adresa experților umani. După cum descoperirea tipăririi cărților a revoluționat propagarea cunoașterii, așa revoluționează sistemele de baze de cunoștințe și sistemele expert valorificarea simplă, ușoară și ieftină a cunoașterii în viața de toate zilele.

Considerăm că un sistem expert este un sistem specializat deoarece:

- se limitează la problemele dintr-un domeniu de specialitate relativ restrâns
- resursele folosite sunt de regulă îmbunătățite pentru rezolvarea performantă a problemelor din domeniul dat, fapt ce-l deosebește de alte sisteme automate destinate prelucrării informațiilor.

Aceste caracteristici le întâlnim și în cadrul “sistemelor dedicate” a căror concepție este cunoscută de mai multă vreme. Din acest punct de vedere considerăm că noțiunea de sistem expert poate fi înțeleasă ca o renaștere a filozofiei sistemelor dedicate pe o treaptă superioară, datorită îmbogățirii calitative a sistemului.

În clarificarea noțiunii de sistem expert trebuie definit în același timp ce anume se înțelege printr-un expert. Expert este o persoană care prin antrenament și experiență, poate îndeplini o sarcină cu un anumit grad de îndemânare care face deosebirea dintre această persoană și alta care nu dispune de această îndemânare sau experiență. Pentru un expert, cunoștințele despre un fenomen sunt organizate, precise, punctuale și sistematizate. Pentru un nespecialist, acestea sunt globale, amorse și nestructurate.

Primele sisteme expert au apărut atunci când s-a amplificat dezvoltarea unor ramuri depășind limitele clasice ale științei, sau la frontiera ei, și care au contribuit cu rezultate deosebite la formarea acestor sisteme. Din acest punct de vedere am putea afirma, deocamdată ca o definiție vagă, că sistemul expert nu este altceva decât un sistem dedicat plus inteligență artificială.

Împărtășind punctul de vedere al altor autori, considerăm că: “Sistemul expert poate fi privit ca materializarea în cadrul sistemului de calcul a bazei de cunoștințe acumulate din experiența expertului, într-o formă în care sistemul este capabil să propună un răspuns inteligent sau să poată realiza o decizie inteligentă într-o situație reală. O altă trăsătură dorită și considerată fundamentală este capacitatea de a explica punctul propriu de vedere sub o formă inteligibilă utilizatorului”⁶⁶.

Sistemele expert înmagazinează cunoștințe specifice prin datele și programele pe care le conțin. Această bază de cunoștințe a sistemului expert este compusă pe de o parte din supoziții (posibilități) specificate exact, pe de altă parte din mulțimea cauzelor, exprimate ca reguli, legate de combinațiile acestor posibilități. Din punct de vedere metodic aceasta se realizează prin înlănțuirea relațiilor de tipul: “Dacă” ... “atunci” ... referitoare la domeniul predefinit al posibilităților. Stabilirea și definirea regulilor este sarcina specialistului din domeniul specificat; realizarea mecanismului pentru structura cunoștințelor și pentru reguli revine specialistului în sisteme expert. Cunoștințele expertului uman trebuie “captate” și reprezentate. Această activitate poartă denumirea de “ingineria cunoașterii” ceea ce nu reprezintă altceva decât analiză de sistem (*system analysis*) la un nivel ridicat.

⁶⁶[***] The British Computer Society’s Group on Expert Systems <http://www.chrisnaylor.co.uk/Definition.html>

Complexitatea sistemelor expert este cu atât mai mare cu cât conțin mai multe reguli. Dacă la început au existat sisteme cu cca. 2.000 de reguli care au fost considerate complexe, azi se pot întâlni sisteme care conțin peste 20.000 de reguli. Pentru realizarea sistemelor expert există din ce în ce mai multe unelte (“tools”) și medii de lucru care permit elaborarea ușoară și rapidă a acestor sisteme.

Sistemele expert pot fi folosite de sine stătător sau pot fi integrate în alte sisteme informatice în funcție de necesități. O clasă specială de sisteme informatice în care sistemele expert pot fi integrate mai ușor din punct de vedere funcțional sunt *Sistemele Interactive pentru Asistarea Deciziilor* (SIAD). Un SIAD⁶⁷ este un sistem informatic care utilizează cunoștințele dintr-un anumit domeniu pentru a ajuta decidentul în rezolvarea unor probleme slab structurate (greu de algoritmat și programat, de exemplu în domeniile: educație, juridic, învățământ).

Cercetările inteligenței artificiale au condus la recunoașterea faptului că dependențele și influențele reciproce dintre fenomenele sociale, economice, psihice sunt cu ordini de mărime mai complicate decât cele întâlnite în legile științelor naturii, și că pentru tratarea lor nu dispunem de utilitate de valoare a celor oferite de analiza matematică pentru descrierea fenomenelor fizice. De aici derivă faptul că sisteme expert existente aproape în totalitate au fost create în domenii tehnico-natural-științifice, deoarece în problemele bazate pe asemenea caracteristici înțelegerea dintre specialiștii de marcă referitoare la existența soluțiilor “corecte” se realizează ușor, majoritatea soluțiilor fiind fundamentate pe legitățile științelor naturii, pe relații de cauzalitate. În situațiile de decizie și în modelele sferei economice respectiv umane, incertitudinea este considerabilă, cunoștințele sunt incomplete; existând mai multe posibilități de comportament și de acțiune pentru alegeri considerate ca “bune”, specialiștii ajung foarte greu la consens în privința stabilirii priorității regulilor.

4.3.1.2. Necesitatea utilizării sistemelor expert

Necesitatea sistemelor expert provine din nevoia de a obține o productivitate sporită în diversele activități desfășurate de experții umani. Un sistem informatic are o capacitate de memorare mult mai mare ca un specialist, care îi permite efectuarea de raționamente identice având o mare capacitate de anticipare. Pe de altă parte, sistemul expert rezolvă o problemă într-un termen foarte scurt față de un specialist căruia îi poate lua mai mult timp pentru a rezolva aceeași problemă.

O problemă de expertiză apare în următoarele situații:

- a) când expertiza este mult prea scumpă
- b) când expertul uman nu poate reține, înțelege sau prelucra rapid volumul mare de informații
- c) când este necesar ridicarea nivelului de pregătire al unor experți
- d) când experții nu sunt disponibili în toate locațiile firmei sau specialistul uman este absent
- e) când avem nevoie de o reducere de personal
- f) pentru expertize de nivel scăzut
- g) când se dorește eliminarea unor puncte de vedere eronate
- h) când datele sunt incomplete sau inconsistente
- i) când expertul uman poate uita unele detalii importante ale problemei sau în cazurile în care ia decizii inconsistente de la o zi la alta
- j) când se dorește să se elibereze timpul experților de vârf pentru a putea fi alocați unor probleme mai complexe
- k) când mai mulți experți sau specialiști sunt necesari în luarea deciziilor iar aceștia nu sunt obișnuiți să-și confrunte punctele de vedere
- l) când expertul uman poate avea prejudecăți în acțiunile sale sau când poate ascunde unele aspecte ale deciziei sale
- m) când expertiza există dar difuzarea cunoștințelor se face anevoios; specialistul este dificil de abordat pentru că este foarte ocupat
- n) când expertul uman evită deliberat asumarea unor responsabilități
- o) când este nevoie de o structurare a cunoștințelor

⁶⁷ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

În toate cazurile se afirmă că introducerea unui sistem expert ameliorează calitatea deciziilor. Sistemele expert deschid largi posibilități raționamentelor prin explorarea unui număr mare de variante, unele chiar nebănuite de către specialiștii în domeniu.

Considerăm că avantajele sistemelor expert în activitatea economică constau în culegerea și prelucrarea rapidă a unui volum mare de date, utilizarea metodelor și modelelor economico-matematice în analiza și interpretarea informațiilor precum și în realizarea unor corelații multiple între elementele și fenomenele caracteristice situațiilor decizionale.

Principalele limite ale sistemelor expert țin de imposibilitatea de a substitui complet și perfect raționamentul uman, de gradul redus în care se iau în considerare incertitudinile mediului economic și de dependența dintre performanțele acestor sisteme software și cele ale sistemelor de calcul pe care sunt implementate. Un alt impediment în utilizarea pe scară largă a acestor sisteme îl reprezintă și costul de elaborare a lor destul de ridicat. Cu toate acestea influența sistemelor expert asupra creșterii eficienței activităților economice în general nu poate fi contestată.

Mai există un aspect în care sistemele expert pot să aducă beneficii considerabile: structurarea cunoștințelor. Atunci când cunoștințele specialistului nu sunt bine modelate sau structurate, singura cale de rezolvare a sarcinilor este dată de sistemul expert. Prin realizarea unui sistem expert se realizează o modelare și o structurare de cunoștințe care nu existau înainte. Această structurare dar mai ales modelare, atunci când este bine făcută, aduce beneficii mari firmei. Un alt câștig apare atunci când se dorește actualizarea cunoștințelor, deoarece este mai simplu să se realizeze actualizarea unor cunoștințe structurate.

Procese economice care se desfășoară astăzi la nivelul firmelor sunt caracterizate de evoluția a două dintre elementele lor definitorii⁶⁸: durata și volumul de informații prelucrate. Pentru a avea drept rezultat o decizie bine fundamentată și oportună, orice proces decizional trebuie să achiziționeze, prelucereze și interpreteze un volum din ce în ce mai mare de informații (mare parte din ele nestructurate), într-un timp din ce în ce mai scurt.

Într-o primă etapă, răspunsul la aceste probleme l-a constituit proiectarea unor sisteme informatice orientate către rezolvarea unei anumite categorii bine determinate de probleme decizionale. Marele dezavantaj al acestora l-a prezentat structura lor compactă care face ca o modificare a informațiilor de intrare sau a metodelor utilizate în rezolvarea problemelor să presupună modificarea integrală a programelor de aplicație. Costurile ridicate aferente modificării produselor soft în vederea cuprinderii tuturor cerințelor viitoare ale utilizatorilor au condus la necesitatea modularizării și standardizării produselor program apoi la separarea cunoștințelor și a metodelor de rezolvare a problemelor de programele utilizator și păstrarea lor într-o bază de cunoștințe. În acest fel, orice modificare apărută în cerințele utilizatorului și în cunoștințele necesare satisfacerii acestor cerințe sau în condițiile externe în care firma își desfășoară activitatea, va influența numai anumite componente ale sistemului informatic, fără să necesite o restructurare completă a acestuia. Soluția în acest sens a constituit-o realizarea unor sisteme de inteligență artificială, utilizate pe scară din ce în ce mai largă la nivelul firmelor odată cu extinderea utilizării tehnicii de calcul în procesele economice.

Foarte mult timp s-a crezut că raționamentul uman în formele sale intuitive, raționament care constituie principalul "motor" în activitatea decizională, nu poate fi încredințat unui calculator. După cel de al doilea război mondial, această convingere a fost infirmată de apariția elementelor de inteligență artificială. Treptat, pe baza acesteia, s-a răspândit ideea că prin folosirea calculatorului s-ar putea realiza tot ceea ce face omul. Cu certitudine, ideea pare încă ambițioasă și este nevoie de foarte multă muncă pentru a se ajunge la performanța de a se înlocui total capacitățile umane cu cele ale calculatoarelor. Nu pot fi ignorate performanțele obținute în rezolvarea problemelor specializate prin utilizarea sistemelor de inteligență artificială iar dacă luăm în considerare volumul cunoștințelor care pot fi stocate și prelucrate în scopul rezolvării anumitor probleme, sistemele bazate pe inteligența artificială prezintă un avantaj net.

În concluzie, dinamica vieții economice și explozia informațională contemporană fac din utilizarea sistemelor expert și în general a sistemelor bazate pe inteligența artificială o necesitate. În ciuda limitărilor impuse de imposibilitatea reproducerii perfecte de către calculator a raționamentelor umane, sistemele expert reprezintă astăzi instrumente indispensabile economistului modern, marea majoritate a activităților decizionale, volumul total al cunoștințelor necesare procesului decizional împreună cu

⁶⁸ [Marinescu98] Marinescu G. -Timpul și deciziile manageriale, Editura Junimea, Iași, 1998.

tehnicele de căutare și regăsire a informațiilor fiind preluate complet de aceste sisteme. În plus, perfecționarea lor continuă, dublată de îmbunătățirea performanțelor computerelor, oferă posibilități sporite de preluare a unor segmente din ce în ce mai cuprinzătoare ale activității de raționare desfășurate de decidentul uman.

4.3.1.3. Caracteristicile sistemelor expert

Cele mai multe programe sunt scrise sub forma unor algoritmi, aceștia conținând practic o mulțime de operații și succesiunea în care trebuie aplicate, utilizând date organizate în fișiere separate sau baze de date externe. Soft-ul de aplicație poate fi privit astfel și ca date+algoritm. De exemplu, un program pentru calculul salariilor poate fi descris în felul următor: caută prima persoană din înregistrări, citește salariul brut al persoanei găsite, calculează deducerile, determină treapta de impozitare în care se încadrează, calculează impozitul, caută toate celelalte scăzăminte și determină salariul net. Toate aceste elemente ajung sub o formă prestabilită pe fluturașul de salarizare și pe statul de plată. După prelucrarea datelor primului angajat, sistemul pentru calculul salariilor continuă determinarea salariilor pentru ceilalți angajați, indiferent de numărul lor. Algoritmul nu se schimbă la nici un angajat. Sistemele de acest gen se bazează pe tehnica de calcul timpurie proiectată pentru rezolvarea problemelor numerice repetitive; aplicațiile mai recente oglindesc deja schimbările tehnologiei. Calculatoarele personale au condus la apariția programelor mai mici de salarizare, bazate pe tabele de calcul, care sunt compuse din șiruri și coloane de numere manipulate cu ajutorul calculatorului.

Oamenii însă nu folosesc numai cuvinte și cifre ci manevrează și informații. Acest fapt a condus la dezvoltarea bazelor de date. Bazele de date înmagazinează o cantitate mare de informații sub formă structurată, utilizatorul având acces la informația dorită oricând, în orice formă dorită. Un exemplu foarte bun pentru ilustrarea celor amintite poate fi baza de date a angajaților. În cadrul acesteia fiecărui angajat îi corespunde un articol, fiecare articol conținând informațiile referitoare la un angajat. Utilizatorul poate obține din informațiile conținute de către baza de date (prin folosirea unui limbaj de interogare) o situație pentru analiză despre angajați, ori de câte ori este nevoie. Poate obține rapoarte de genul: lista angajaților cu salariul brut peste 8.000.000 lei, cu vechime peste 3 ani, domiciliați într-un anumit oraș etc. Ceea ce putem obține din baza de date este limitat doar de către informația înmagazinată și de către limbajul de interogare.

La baza programelor “clasice” de calculator stă conceptul de algoritm, care de cele mai multe ori implică o organizare secvențială, predefinită de operații. Aceste programe însă par să fie incapabile să rezolve probleme în care intervin raționamente, decizii sau în care pot să apară situații ce nu se pot specifica apriori. Din această cauză, destul de mult timp s-a crezut că raționamentul și comportamentul uman nu poate fi modelat prin sisteme informatice. Ne putem imagina cât de dificilă este înțelegerea propoziției: “Am văzut Statuia Libertății zburând deasupra New-York-ului”⁶⁹. Ea va trebui să fie interpretată în sensul că interlocutorul zbura și nu statuia. De aceea pentru înțelegerea limbajului uman este necesar un volum foarte mare de cunoștințe.

Ideea modelării raționamentului și comportamentului uman este foarte ambițioasă însă ea reprezintă obiectivul urmărit azi de cercetătorii din domeniul inteligenței artificiale. Cercetările acestora se desfășoară în diverse direcții, cum ar fi: rețele neuronale, sisteme solvitoare, baze de cunoștințe, sisteme expert, recunoșterea formelor, recunoșterea și reproducerea limbajului natural, robotica etc. Pe lângă acestea nu trebuie uitate cercetările fundamentale care privesc sistemele cognitive, reprezentarea și manipularea cunoștințelor.

“Ne scâldăm în informații dar suntem avizi de cunoaștere” spunea John Naisbitt. Cunoașterea însă este mai mult decât simpla informație, ea reprezintă, pe lângă valoarea pur matematică și înțelegerea informației respective. Cunoașterea adaugă la informație un înțeles, asociază o acțiune. Omul se interesează nu numai de informații dar și de cunoștințe. Informația este mulțimea datelor organizate comprehensibil pe când cunoașterea este informația definită din punctul de vedere al aplicației date.

În literatura de specialitate există încă discuții aprinse despre ce sunt cunoștințele și care sunt diferențele dintre date și cunoștințe. Majoritatea însă clasifică cunoașterea în două categorii: cunoaștere explicită și tacită. Cunoașterea explicită este clar afirmată, definită și poate fi comunicată altora pe când

⁶⁹ [***] Language Understanding and Acquisition <http://home.ku.edu.tr/~dyuret/pub/yuretphd/node1.html>.

cea tacită presupune cunoștințe care nu sunt afirmate dar sunt implicit presupuse, sunt dificil de extras și de împărtășit celor din jur. Cunoașterea explicită poate fi gestionată mai ușor deoarece există în forme tangibile (cărți, manuale sau pe diverse suporturi electronice) pe când cea tacită trebuie convertită în cunoaștere explicită pentru a putea fi gestionată.

Cunoștințele reprezintă, pentru programarea declarativă ceea ce au reprezentat datele în programarea clasică. Sistemele expert reprezintă tranziția de la procesarea datelor la procesarea cunoștințelor.

Figura nr. 20 ilustrează diferențele între procedurile aferente prelucrării datelor și procesarea cunoștințelor în vederea luării deciziilor. În procesarea tradițională a datelor, decidentul obține informațiile de care are nevoie și face o analiză explicită a lor înainte de a lua decizia.

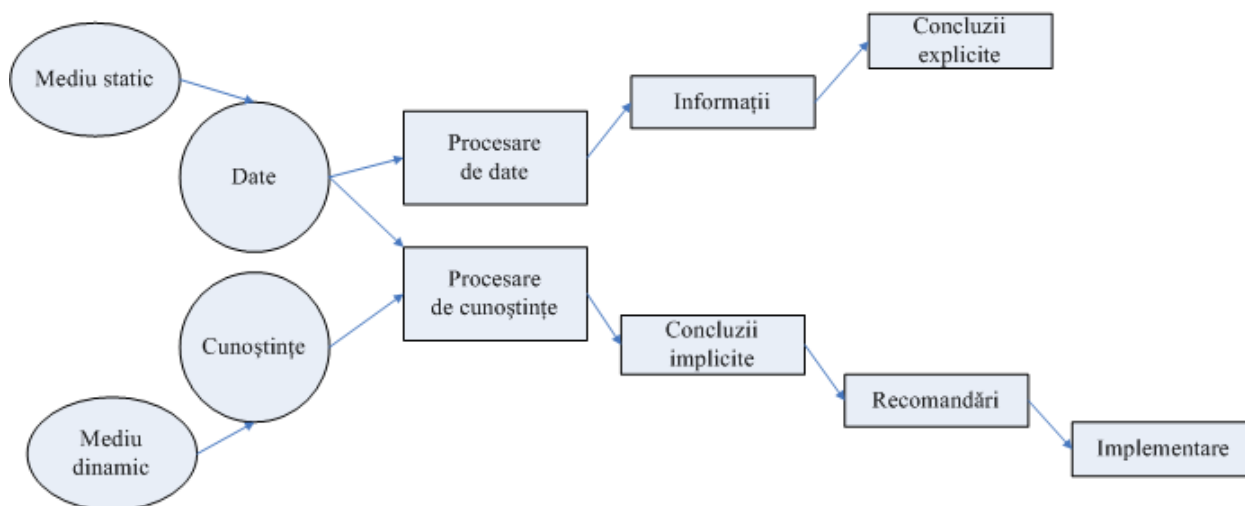


Fig 20 Relația procesare date –procesare cunoștințe

Pe baza celor prezentate anterior, considerăm că diferențele dintre sistemele expert și programele tradiționale pot fi redate pe baza următoarelor caracteristici:

<i>Pachetele de programe tradiționale</i>	<i>Sistemele expert</i>
-manevrează date -prelucrare numerică -aplică algoritmi -folosește prelucrări repetitive -se clădește pe baze de date -funcționează cu un set complet de date	-manevrează cunoștințe -prelucrare simbolică -aplică metode euristice și reguli -folosește inferența -se clădește pe baze de cunoștințe -funcționează și cu câteva reguli (ca și prototip)

Sistemele expert ce rulează pe calculatoarele personale au fost create pentru colectarea, înmagazinarea, interogarea și realizarea accesibilității cunoașterii dintr-un anumit domeniu.

Sistemul expert poate fi privit ca materializarea în cadrul calculatorului a bazei de cunoștințe acumulate din experiența expertului, într-o formă în care sistemul este capabil să propună un răspuns inteligent sau să poată realiza o decizie inteligentă într-o situație reală.

Sistemele expert, încercând să imite expertul uman, posedă următoarele caracteristici:

- cunoștințele sunt independente de mecanismul deductiv;
- ele se introduc în forma brută și nu influențează mecanismul deductiv;
- mecanismul deductiv (raționamentul) se bazează pe prelucrarea dinamică a bazei de cunoștințe conform unor reguli și nu pe un algoritm matematic, specific programării clasice (procedurale);
- rezultatele expertizei nu sunt cunoscute anterior și sunt imprevizibile;
- gestionează baze de cunoștințe de volum mare și sunt capabile să trateze cunoștințe inexacte și incomplete;
- utilizează metode empirice care au la bază experiența experților, fapt care conduce la obținerea celor mai bune soluții;
- sunt specializate pentru a rezolva probleme dintr-un anumit domeniu și nu a unei singure probleme ca în programarea clasică (bazată pe algoritmi);
- cunoștințele manipulate sunt de natură simbolică, declarativă, în opoziție cu programarea clasică care operează cu informații numerice;
- sistemele expert explică raționamentele făcute și argumentează, asemenea unui expert uman, soluțiile obținute.

Waterman⁷⁰ arată că sistemele expert au patru atribute de bază:

1. *Expertiza* –conferă unui sistem expert următoarele caracteristici:
 - performanță de expert
 - un nivel ridicat de calificare
 - o robustețe adecvată
2. *Raționament simbolic* –permite sistemelor expert:
 - folosirea reprezentării simbolice a cunoașterii
 - reformularea cunoașterii simbolice
3. *Profundzime* –asigură sistemelor expert:
 - abordarea problemelor dificile
 - utilizarea cunoașterii complexe și mecanismelor inferențiale pentru prelucrarea ei
4. *Autocunoaștere* –determină:
 - examinarea propriilor raționamente
 - explicarea operațiilor și acțiunilor
 - favorizează învățarea

Conform lui Joseph Giarratano și Gary Riley⁷¹, un sistem expert are următoarele caracteristici: -înalță performanță în calitatea concluziilor

⁷⁰ [Waterman86] Waterman D.A. –A Guide to Expert Systems, Addison-Wesley, Reading, 1986.

⁷¹ [Giarratano93] Giarratano J., Riley G. –Expert Systems. Principles and Programming, Designing Expert Systems using CLIPS, PWS Pub, Boston, 1993

-timp de răspuns adecvat

-înalță fiabilitate

-flexibilitate sporită în gestiunea cunoștințelor

Frank Puppe (Universitatea din Karlsruhe) consideră că un sistem expert se caracterizează prin:

-tranparență (explicarea soluției oferite)

-flexibilitate în gestiunea cunoștințelor

-competență în soluționarea problemelor

Pentru a contura noțiunea de sistem expert putem enumera următoarele caracteristici:

- din punct de vedere conceptual, vizează reconstituirea raționamentului uman pe baza expertizei obținute de la experți;
- dispun de cunoștințe și de capacitatea de a desfășura activități intelectuale umane;
- sunt organizate pentru achiziția și exploatarea cunoașterii dintr-un domeniu particular numit domeniul problemei;
- dispun de metode de invocare a cunoașterii și exprimarea expertizei comportându-se ca un „asistent inteligent”;
- la nivel de realizare informatică, sistemele expert se bazează pe principiul separării cunoașterii de programul care o tratează;
- sunt capabile să memoreze cunoașterea, să stabilească legăturile dintre cunoștințe și să infereze concluzii, soluții, recomandări.

O problemă care se pune în privința folosirii sistemelor expert este aceea a raportului dintre relevanță și precizie⁷². Se știe din teoria sistemelor că un sistem descompus într-un număr mic de subsisteme are o relevanță mare și o precizie mică, iar pe măsură ce descompunerea avansează, relevanța scade și precizia crește. Prelucrarea algoritmică se caracterizează printr-o precizie mare, iar sistemele expert printr-o relevanță mare, pe care o preia din practica rezolvării problemelor, ceea ce face ca acesta să fie un mijloc important pentru dominarea complexității. Se pot oferi soluții mai productive decât programele algoritmice în situațiile în care cerințele informaționale se modifică des. Sistemele expert permit de asemenea extinderea ariei informaticii spre domenii greu de algoritmat, cu informații preponderent calitative: educație, politic, juridic.

4.3.1.4. Structura unui sistem expert

Se știe⁷³ că expertul uman raționează pe baza memoriei de lungă durată, în care își înmagazinează cunoștințele de specialitate pe baza raționamentelor, care reprezintă un mecanism general. Analog expertului uman,^{74 75} sistemele expert dispun de o bază de cunoștințe (BK), care este compusă din fapte, adică predicate sau clauze exact specificate și de un motor inferențial (MI) care conține reguli, legate de combinațiile acestor fapte. (BK) și (MI) sunt independente; un (MI) poate fi utilizat cu diverse (BK), sistemele obținute comportându-se ca sisteme expert diferite. Astfel se explică utilizarea motorului inferențial al sistemului MYCIN, specializat în aplicații medicale, pentru aplicații de afaceri. De asemenea sistemele expert dispun de un modul explicativ.

În majoritatea lucrărilor de specialitate se consideră că principalele componente ale unui sistem expert sunt următoarele⁷⁶:

1. Baza de cunoștințe conține ansamblul de cunoștințe specializate într-un anumit domeniu, preluate de cognitivian de la expertul uman. Privite în mod general, cunoștințele reflectă “obiectele” lumii reale și relațiile dintre ele. În literatura de specialitate se consideră că există mai multe metode de reprezentare a cunoștințelor, dintre care cele mai importante sunt: rețelele semantice, regulile de producție

⁷² [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

⁷³ [Boden87] Boden M.A. -Artificial Intelligence and Natural Man, MIT Press, 1987.

⁷⁴ [Goonatilake95] Goonatilake S. -Intelligent Systems for Finances and Business: An Overview, John Wiley&Sons, 1995.

⁷⁵ [Parsaye89] Parsaye K. -Intelligent Databases, John Wiley, 1989.

⁷⁶ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

și cadrele. Procesul de creare a bazei de cunoștințe, care constă în preluarea acestora de la expertul uman, modelarea de către cognitician în conformitate cu cerințele metodei de reprezentare, introducerea în bază și validarea, este destul de laborios și necesită o colaborare permanentă între cognitician și expert. Acest proces presupune foarte multe iterații și teste în cursul cărora însăși expertul uman poate fi pus în dificultate pentru argumentarea unor opțiuni.

Baza de fapte conține datele unei probleme concrete care urmează să fie rezolvată precum și faptele rezultate în urma raționamentelor efectuate de motorul de inferență asupra bazei de cunoștințe. În unele lucrări, baza de fapte este inclusă în baza de cunoștințe. Totuși se consideră că trebuie făcută distincție între cunoștințe, care descriu rezolvarea unei clase de probleme și fapte, care descriu aserțiuni de instanțiere a unei probleme din clasa respectivă. Faptele sunt cunoștințe elementare, invariabile, conținute în aserțiuni sau declarații.

Baza de reguli conține totalitatea regulilor aplicabile faptelor pentru a se obține raționamente. Regulile reprezintă un tip de cunoștințe constante care organizează faptele, în sensul stabilirii unor relații între acestea. O regulă reprezintă o asociere între descrierile de stare ale unei probleme și acțiunile de întreprins în cazul în care problema se află într-o anumită stare. În general, o regulă se compune dintr-o parte stângă, exprimând o condiție și o parte dreaptă, conținând concluzii.

2. Motorul de inferență este elementul efectiv de prelucrare în sistemele expert, care pornind de la fapte (datele de intrare ale problemei) activează cunoștințele corespunzătoare din baza de cunoștințe, construind astfel raționamente care conduc la noi fapte (de ieșire).

Inferența (din latinul *inferre* a duce) este o operație logică de derivare a unui enunț din altul, prin care se admite o judecată (al cărui adevăr nu este verificat direct) în virtutea unei legături a ei cu alte judecăți considerate adevărate.

Motorul de inferență construiește un plan de rezolvare în funcție de specificul problemei, utilizând cunoștințele din domeniul respectiv. În urma acțiunii motorului de inferență, baza de cunoștințe se îmbogățește fie prin adăugarea unor elemente noi, fie prin modificarea celor existente.

3. Modulul explicativ are rolul de a prezenta într-o formă larg accesibilă (limbaj natural) rezultatul raționamentelor efectuate de motorul de inferență și totodată întrebările la care trebuie să răspundă utilizatorul. Acest modul este util și expertului uman pentru verificarea coerenței bazei de cunoștințe. Utilizatorul sistemului expert poate pune întrebări de genul ce, cum și de ce în legătură cu aspectele unei raționări.

4. Modulul de achiziție a cunoștințelor are rolul de a transforma cunoștințele din forma în care le exprimă cogniticianul (inginerul de cunoștințe) în forma internă de memorare pe suport.

5. Interfața cu utilizatorul realizează dialogul utilizatorului cu sistemul expert, în sensul specificării datelor de intrare și al furnizării rezultatelor.

Arhitectura unui sistem expert este ilustrată în figura 21.

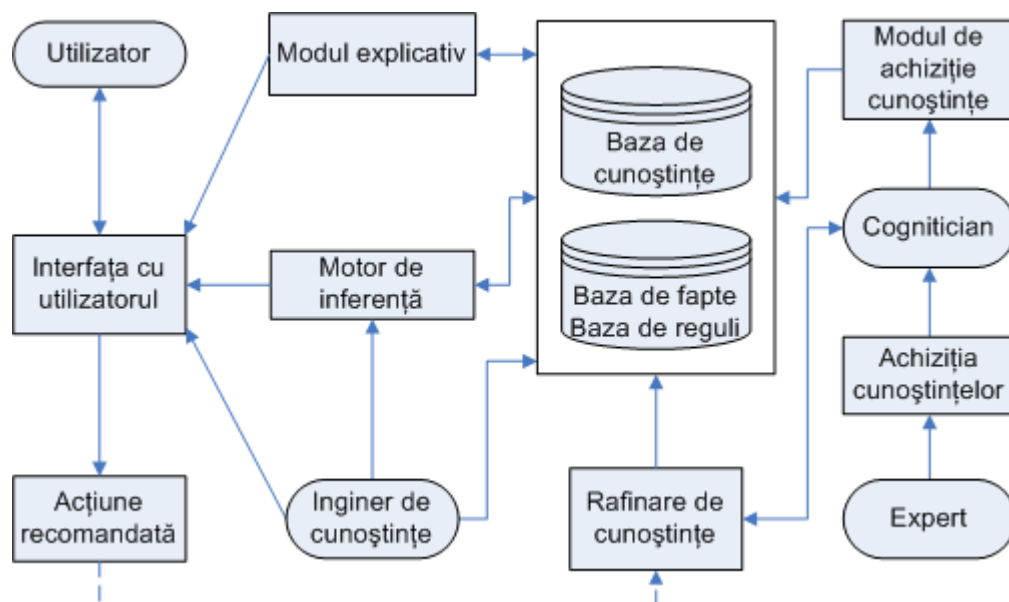


Fig. 21 Arhitectura unui sistem expert

4.3.1.5. Tipuri de sisteme expert

După modul de reprezentare a cunoașterii, avem:

-Sisteme expert bazate pe reguli

-Sisteme expert bazate pe cadre

-Sisteme hibride –combină mai multe metode de reprezentare a cunoașterii.

Asupra acestora vom insista în subcapitolul destinat reprezentării cunoașterii.

După natura problemelor soluționate, sistemele expert se împart în⁷⁷:

-Sisteme expert cu colectarea faptelor doveditoare –acestea sunt folosite în expertizele contabile, tehnice, medicale

-Sisteme expert cu rafinare în pași succesivi –acestea conțin un număr mare de ieșiri obținute prin detalierea succesivă pe niveluri a problemei din domeniu

-Sisteme expert cu asamblare în pași succesivi -la acestea, ieșirile într-un număr foarte mare au legătură cu o paletă largă de rezultate posibile

-Sisteme expert “la cheie” -sunt elaborate la cerințele specifice ale unui utilizator

-Sisteme expert în timp real –sunt folosite în general în cadrul unor procese industriale (dar nu numai) care necesită un timp de răspuns foarte strâns necesar luării unei decizii.

După scopul elaborării lor, sistemele expert pot fi clasificate în mai multe tipuri:

-Sisteme expert de tip diagnostic -presupun realizarea unor expertize, de multe ori bazate pe reguli euristice (sisteme de supraveghere în timp real a unor procese, diagnosticarea unei întreprinderi din punct de vedere al activității financiare etc) fiind cele mai utilizate în practică. Dintre acestea, cel mai cunoscut este MYCIN (elaborarea unui diagnostic preliminar în cazul infecțiilor bacteriene ale sângelui).

- Sisteme expert pentru reparații –sunt folosite în aplicațiile de reparare, oferind diagnostice pentru care dezvoltă un plan de reparații și controlează decizia.

-Sisteme expert pentru instruire –sunt utilizate în procesul de învățământ. În cadrul acestora, cei care se instruiesc pot pune întrebări ca la un expert uman. Un exemplu este CADHELP (instruire pentru proiectarea asistată de calculator).

-Sisteme expert de interpretare –se utilizează în procesul de analiză a informațiilor pentru a le determina semnificația. Cel mai cunoscut este DENDRAL folosit pentru interpretarea structurii moleculare.

⁷⁷. [Andone99] Andone I. Țugui A. –Sisteme inteligente în management, contabilitate, finanțe-bănci, Editura Economică, București, 1999.

-Sisteme expert de prognoză –sunt folosite în predicție pentru a determina viitoarele condiții din anumite situații.

-Sisteme expert pentru proiectare și planificare –sunt folosite în general în aplicații ingineresti și de management dar nu numai. Un exemplu este sistemul SPEX care permite experimentări în biologie moleculară.

-Sisteme expert de control sunt utilizate în scopul sprijinirii unor decizii ce trebuie luate rapid (în timp real) și care prelucreează un volum mare de date. Acestea pot fi deseori sisteme de reglare și control al unui proces.

După rolul lor, avem următoarea clasificare:

-Expert autonom -sistemul analizează o situație, ia decizii acționând în consecință, comandând acțiuni fără a avea nevoie de supervizare umană. Acestea se aplică în cazul unor diverse procese industriale în care modificarea unor parametri (temperatură, presiune) necesită anumite acțiuni specifice, viteza de reacție fiind crucială. De exemplu la centralele nucleare, numeroase circuite sunt automatizate, în caz de avarie coordonarea fiind făcută de astfel de sisteme datorită numărului mare de variabile și a timpului scurt de răspuns necesar.

-Expert autonom cu supervizare umană –acestea funcționează ca un sistem autonom dar prezintă concluziile unui expert uman. Sunt în general sisteme care oferă alarme în cazul unor disfuncționalități ale unor procese, deciziile fiind apoi luate de experți umani.

-Consultant –sunt sisteme care, pe baza datelor introduse sau culese din mediul în care își desfășoară activitatea, oferă expertize la cererea unui utilizator, căruia astfel i se îmbunătățesc performanțele, apropiindu-l de un expert în domeniu.

-Coleg sau asistent inteligent –sistemul construiește seturi de sugestii pe care le oferă unui coleg uman. Acest tip cuprinde o mare parte din sistemele expert utilizate în economie. Ele pot analiza un număr mare de variante, expertul uman alegând-o pe cea care i se pare optimă. Astfel se pot lua decizii privind oportunitatea unor investiții, credite bancare etc. În domeniul producției, poate determina diferite combinații de valori ale parametrilor furnizați de senzori care pot constitui probleme pentru fluxul de producție.

-Asistent de nivel coborât –acestea execută sarcini mari consumatoare de timp, eliberându-l pe utilizator de unele părți dificile ale sarcinii sale.

4.3.1.6. Funcțiile unui sistem expert

Principalele funcții ale unui sistem expert așa cum sunt prezentate și de Benchimol⁷⁸ sunt următoarele:

- Funcția de interpretare: permite traducerea de semnale provenite din captarea de exemple sau de date brute în expresii simbolice care pot fi folosite în raționamente;
- Funcția de diagnostic: asigură stabilirea unei corelații între caracteristici sau simptome și situații tip;
- Funcția de formare: permite transmiterea de cunoștințe unui elev a cărui nivel și caracteristici fac obiectul unui diagnostic de învățare foarte bine adaptată; această transmitere de cunoștințe se poate baza pe diagnostic, întreținere, concepție etc;
- Funcția de supraveghere: asigură activarea unui sistem de alarmă în condiții determinate; acest semnal se bazează pe anumite stări interpretate și folosite în stabilirea unui diagnostic
- Funcția de previziune: permite anticiparea unei situații, plecând de la o stare curentă, în general prin intermediul unui model construit pe o bază istorică sau prin învățare;
- Funcția de simulare: permite deducția, plecând de la un model, a consecințelor de acțiune sau evenimente declanșate de către sistemul în curs de derulare a simulării;
- Funcția de planificare: permite definirea în timp și în spațiu a acțiunilor care permit să se atingă o stare finală prin compararea stării curente cu starea dorită ținând cont de consecințele acestei acțiuni, respectând anumite reguli și acțiuni;

⁷⁸ [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.

- Funcția de întreținere: decurge dintr-un diagnostic arătând care sunt punctele slabe ale unui sistem, indicând și cauzele; planul de acțiune constă în furnizarea instrucțiunilor necesare pentru efectuarea remedierii;
- Funcția de concepție: creează mulțimea de decizii care permit, ținând cont de performanțele fixate, ca urmare a unui diagnostic, să se determine sarcinile ce trebuie îndeplinite, în funcție de nevoile exprimate la un moment dat, și furnizarea unor mijloace pentru a se atinge acest obiectiv;
- Funcția de control și pilotaj: creează mulțimea de acțiuni aplicate la un sistem după informațiile care rezultă din supravegherea sistemului și din anticiparea situațiilor care vin să asigure printr-o întreținere permanentă și un răspuns adaptat la diverse situații, o funcționare a sistemului apropiată de cea mai posibilă funcționare normală.

Într-un sistem real, aceste funcțiuni de bază sunt adesea combinate între ele.

La ora actuală, funcțiunile cele mai utilizate sunt: cea de diagnostic (în jur de 45% din aplicații), de concepție și planificare (20%), de interpretare (20%) și de control / pilotaj / supraveghere (15%). În domeniul economic și financiar funcțiile de diagnostic și de formare a vânzătorilor par să fie în centrul preocupărilor elaboratorilor (diagnosticul întreprinderilor, creditul întreprinzătorilor etc). În elaborarea aplicațiilor în domeniul asistării deciziilor, considerăm că funcțiile de planificare, diagnostic și de simulare se pot combina.

4.3.1.7. Roluri în cadrul sistemului expert

La conceperea și realizarea unui sistem expert trebuie să participe următoarele categorii de personal⁷⁹:

1. experții umani ale căror cunoștințe urmează a fi colectate și transferate sistemului, urmând apoi să fie difuzate către utilizatori prin intermediul sistemelor expert

2. cogniticienii care asigură transpunerea cunoștințelor și strategiilor de raționament ale expertului uman în structurile specifice metodei de reprezentare a cunoștințelor și a instrumentelor informatice utilizate.

3. inginerii cunoașterii aceștia sunt cei care asigură rezolvarea problemelor informatice implicate de realizarea sistemului expert. Ei pot coincide cu cei care îndeplinesc funcția de cognitician (cu condiția de a avea cunoștințele și capacitatea profesională necesare).

4. utilizatorii finali sunt cei care vor exploata sistemul valorificând cunoștințele încorporate în el.



Fig. 22 Roluri în cadrul unui sistem expert

Figura nr. 22 ilustrează rolurile existente în cadrul unui sistem expert. Aportul acestor categorii în elaborarea sistemelor expert poate fi concretizat astfel:

Expertul uman (*domain expert*) este acel specialist uman documentat care (posedând cunoștințe profunde și experiență vastă) reprezintă sursa de informații pe cunoașterea căreia se clădește baza de cunoștințe. În unele cazuri se poate întâmpla ca un articol științific sau o carte de specialitate să poată

⁷⁹ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

înlocui expertul uman al domeniului. De regulă, acesta reprezintă chiar subiectul dorit a fi substituit prin serviciile oferite de-a lungul consultării sistemului expert realizat.

La generatoarele de sisteme expert, integrarea interfeței utilizator și a mașinii de inferență în mediu poate ajuta expertul domeniului să devină el însuși autorul bazei de cunoștințe.

Cogniticianul (*knowledge base author* -autorul bazei de cunoștințe sau inginerul de cunoștințe) este persoana care pe baza cunoștințelor expertului domeniului realizează baza de cunoștințe. Termenul cognitician definește o funcție specifică și în general indispensabilă în realizarea unui sistem expert și nu o persoană fizică. Adesea, cogniticianul este de fapt o echipă, dar indiferent dacă acesta este singur sau nu, se pot distinge mai multe aspecte în munca sa: ingineria generală a cunoștințelor, structurarea și extragerea cunoștințelor. Atunci când un cognitician comunică cu un expert, el se găsește în fața unei probleme de înțelegere. Vocabularul folosit de expert este inadecvat pentru a traduce demersul său și de aceea este nevoie de o perioadă de adaptare. Cogniticianul trebuie să dea dovadă de răbdare, să fie capabil de un excelent dialog pentru a putea intra într-un domeniu care nu-i aparține, să stăpânească tehnica interviurilor. Sarcina cea mai delicată a cogniticianului este structurarea cunoștințelor expertului, identificarea și formalizarea metodelor de raționament.

Munca referitoare la achiziția cunoștințelor este fundamentală pentru calitatea finală a sistemului. S-a constatat că persoanele care au reușit în realizarea acestor sarcini a avut profiluri și formații foarte diferite și că nu este neapărat necesar să fie foarte specializați în informatică. Considerăm că, printre calitățile necesare unor cogniticieni trebuie să se enumere:

- să fie pedagogi, psihologi, să aibă simțul relațiilor umane și al comunicării;
- să cunoască metodele inteligenței artificiale, sistemele expert, reprezentarea cunoștințelor;
- să fie curioși pentru a se interesa de problema ce face obiectul sistemului expert; să fie cultivați și să aibă o bogată experiență de viață pentru a putea discuta cu comanditarul sistemului, cu experții astfel ca la sfârșitul procesului informațiile culese să-i fie utile;
- să fie creativi și să aibă imaginație pentru a fi capabili să conceapă baza de cunoștințe, pornind de la o situație complexă nouă, pentru care nu se găsește un model al tuturor faptelor.

Aceste calități corespund unor profesii diferite de grade diferite ceea ce face ca să fie recomandată munca în echipă. Diversitatea acestor calități explică tocmai de ce nu există profesioniști privilegiați pentru a deveni cognitician.

Inginerul cunoașterii (*knowledge engineer*) este acel specialist în sisteme informatice și tehnică de calcul care proiectează și realizează generatorul de sisteme expert. Activitatea acestuia este foarte importantă deoarece astfel avem posibilitatea de a cunoaște sistemele expert și de a realiza aceste sisteme fără a ne afunda într-un cod de programare de nivel inferior. Inginerul cunoașterii alcătuiește deci celelalte component importante ale sistemului expert: interfața utilizator și mecanismul de inferență. Inginerii cunoașterii trebuie să stăpânească perfect, pe de o parte limbajul de programare ales pentru realizarea sistemului (LISP, PROLOG, etc) iar pe de altă parte teoria, practica și natura sistemelor expert. Ei pot fi identici, ca persoane, cu cei care îndeplinesc funcția de cognitician, evident cu condiția de a avea cunoștințele și capacitatea necesare proiectării și realizării sistemului expert. Prin facilitățile oferite de generatoarele de sisteme expert, efortul de programare este mult mai redus, uneori aproape inexistent.

Rolul principal aparține utilizatorului (*user*) pentru că el este cel pentru care s-a propus rezolvarea problemei sau atingerea scopului și așteaptă răspunsul sistemului expert. Sistemul adresează pe rând întrebări utilizatorului, sistematizează informațiile cele mai importante primite, explică neobosit rostul întrebărilor puse și scopul investigației. În final, după furnizarea recomandărilor referitoare la soluționarea problemei utilizatorului sau după atingerea scopului propus de către acesta, sistemul va motiva (în profunzimea solicitată de utilizator) și va explica de ce a furnizat recomandările comunicate. Dialogul dintre sistem și utilizator trebuie să fie prietenos, iar limbajul folosit trebuie să fie adaptat la tipul de problemă și la utilizator, să urmeze nivelul și rolul jucat de acesta.

4.3.1.8. Baza de cunoștințe

De unde știe omul anumite lucruri? Iată o întrebare la care se poate răspunde relativ simplu: înmagazinăm foarte multe fapte, iar unele dintre acestea se leagă de situația în care ne aflăm. De aici apare și următoarea întrebare: de unde știm care sunt faptele care se leagă și cum să le aplicăm? Memoria umană conform unei viziuni simpliste se compune din cutii sau casete adresabile. Fiecare casetă are un

conținut și o etichetă. De exemplu cea cu eticheta “zi pentru activități didactice” conține cunoștințele referitoare la zilele în care desfășurăm activități didactice. Acest mod de organizare conține însă o greșeală: se poate întâmpla ca gândurile sau o discuție referitoare la o anumită temă –de exemplu referitoare la mâncăruri- să genereze gânduri despre tema respectivă în locul celei inițiale. Ca urmare orice zonă din memoria noastră poate activa o altă zonă, atât timp cât există o legătură (fie ea cât de slabă) între ele.

După un model mai sofisticat, se consideră că memoria noastră este colecția extraordinară a informațiilor și cunoștințelor. Nu ajunge să posedăm cunoștințe, trebuie să știm și la ce anume se referă ele (adică trebuie să cunoaștem ce anume știm). Cunoașterea modului de organizare a cunoștințelor se numește cunoaștere adâncă. Conținutul informațional este important dar mai importantă este relația dintre conținuturi. Aceste cunoștințe, înmagazinate sub forma unor reguli, se cheamă *euristice*. Cuvântul provine din limba greacă, desemnând metoda educațională prin care învățăceii descoperă de la sine răspunsurile la problemele date și are aceeași rădăcină ca și cuvântul *evrika* ce înseamnă a descoperi.

Memoria umană poate fi privită deci ca o mulțime de cunoștințe euristice care nu pot fi identificate și definite ușor, dar dacă dorim crearea unei baze de cunoștințe pentru un sistem expert trebuie să procedăm mult mai exact. Aceste cunoștințe se înmagazinează sub forma unor reguli care conțin perechi condiție-eveniment (dacă condiția este adevărată atunci se realizează operația, de exemplu:

*Dacă plouă,
Atunci voi lua o umbrelă cu mine.*

Condițiile pot fi exprimate în diferite moduri: prin afirmații (exemplu: dacă plouă), comparări numerice (=,< etc), pot fi aplicate asupra unor simboluri al căror conținut este colectat direct din lumea exterioară, pot fi folosite diverse combinații (folosind operatori logici etc). Operațiile, evenimentele pot fi declarații, instrucțiuni și pot determina dirijarea evenimentelor lumii exterioare (de exemplu cazul dirijării producției). Orice declarație poate fi folosită pentru activarea altei reguli. În afara perechii condiție-eveniment, fiecare regulă mai are două părți. În primul rând trebuie să dispună de o etichetă. Această etichetă poate fi una simplă (de genul “*Regula R1*”) sau una explicativă (de exemplu regula “*Ia-ți umbrela*”). Evidența regulilor este simplă într-un sistem compus din zece reguli, dar această preocupare devine din ce în ce mai greu de urmărit în cazul creșterii dimensiunii bazei de reguli. Din această cauză multe pachete de generare dispun de dicționare de reguli, care permit autorului să marcheze anumite drumuri prin referirea regulilor considerate. Ca exemple pentru ilustrarea ordinelor de mărime: un sistem demonstrativ poate fi compus din aproximativ 50 de reguli; un sistem funcțional poate dispune de 1.000 de reguli în timp ce un sistem mare poate conține până la zeci de mii de reguli. Un om poate deține –după unele aproximări- circa 100.000 de reguli (inclusiv euristice). Deși sistemele expert pot fi foarte mari, și un sistem redus cu 50 de reguli se poate dovedi foarte util.

O altă componentă importantă a regulilor este *explicația*. Nivelul explicativ este una din diferențele cele mai evidente dintre sistemele expert și programele tradiționale. Sistemele expert sunt capabile într-o anumită măsură să-și explice comportamentul, dar explicațiile trebuie clădite în sistem. Sistemele expert care există azi nu sunt capabile să realizeze propriul sistem ajutător pe baza regulilor din baza de cunoștințe. O regulă completă ar putea fi următoarea:

Etichetă:	<i>R1 (luarea umbrelei)</i>
Condiție:	<i>Dacă plouă</i>
Acțiune:	<i>Atunci luați umbrela</i>
Explicație:	<i>Folosind umbrela puteți evita să vă udați</i>

Se poate afirma că lumea nu poate fi descrisă simplu printr-un set de reguli. Fără îndoială însă, priceperea se materializează de obicei prin reguli. O mare parte a activității noastre umane poate fi privită ca un ansamblu de reguli sau elemente euristice și poate fi încorporată în baza de cunoștințe a unui sistem expert. Euristica nu garantează rezultatele ca fiind sigure, așa cum o fac algoritmii convenționali care sunt prezenți în sistemele de asistare a deciziilor dar ele oferă în majoritatea cazurilor rezultate care sunt suficient de corecte pentru a fi folosite.

4.3.1.9. Achiziția cunoștințelor

Prin intermediul acestui proces se determină care sunt tehnicile, cunoștințele și euristicile folosite de către experți pentru a realiza expertizele în domeniu. Achiziția de cunoștințe de la experții umani se numește *ingineria cunoașterii* și poate fi împărțită în următoarele faze:

- *Identificare* –definirea corectă a problemei și determinarea caracteristicilor sale
- *Conceptualizare* –determinarea conceptelor care să sprijine reprezentarea cunoștințelor
- *Formalizare* –alegerea unor metode de reprezentare a cunoștințelor
- *Implementare* –reprezentarea propriu-zisă a cunoștințelor în formalismul ales
- *Testare* –verificarea cunoștințelor și validarea bazei de cunoștințe

Ilustrăm în figura 23 fazele procesului de achiziție a cunoștințelor.

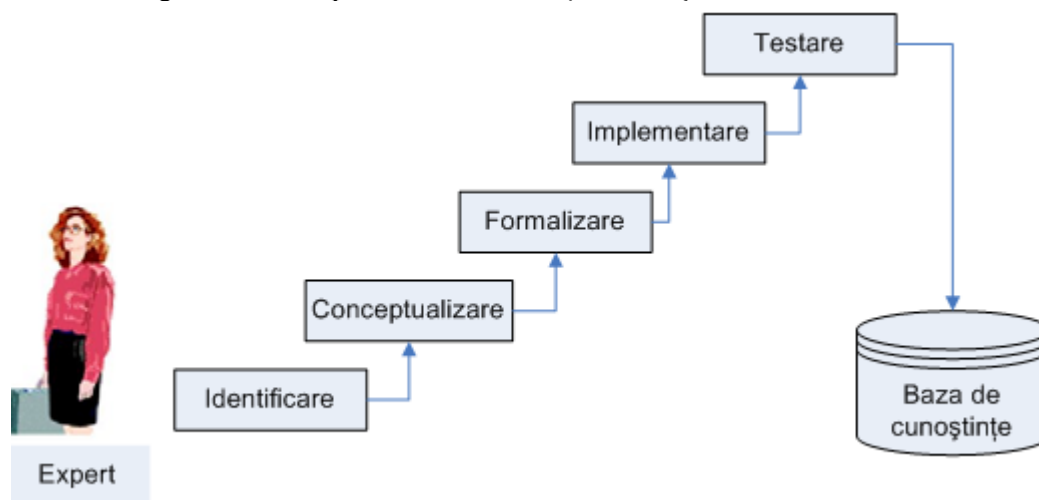


Fig. 23 Procesul de achiziție a cunoștințelor

Una din cele mai importante etape ale acestui proces o reprezintă conceptualizarea care permite modelarea cunoștințelor utilizate prin crearea unei structuri conceptuale utilizate de către expert în domeniul său de activitate. În această fază este important modul de comunicare dintre expert și cognitivian, pentru că trebuie definite toate noțiunile care vor fi folosite, pașii care vor fi efectuați în elaborarea expertizei și restricțiile care apar în rezolvarea problemei. Efortul de conceptualizare este necesar pentru a determina cât mai exact domeniul de expertiză al sistemului expert și a fundamenta formularea cunoștințelor în cadrul bazei de cunoștințe.

În analiza conceptuală, considerăm că trebuie evidențiate următoarele aspecte:

- identificarea principalelor “obiecte”, concepte sau entități din domeniul de expertiză
- definirea atributelor necesare pentru descrierea obiectelor, conceptelor sau entităților
- identificarea celor mai importante corelații care există între acestea
- identificarea restricțiilor care intervin în relațiile dintre obiecte
- specificarea tipurilor de probleme la care urmează să răspundă viitorul sistem
- definirea cadrului general de utilizare al sistemului expert, atât sub aspectul manierei de comunicare a problemelor de rezolvat (dialog de la terminal, consultarea unor baze de date etc) cât și al interacțiunii cu utilizatorul final.

După analiza conceptuală urmează efectiv colectarea cunoștințelor. Apar anumite aspecte specifice acestei faze, cum ar fi:

- lipsa unei sistematizări inițiale, deoarece expertul în timpul desfășurării activității nu este nevoit să-și exteriorizeze cunoștințele
- tendința de a omite cunoștințele generale aparținând profesiei (așa numitele cunoștințe de “bun-simț”) care pentru expertul uman nu sunt subînțelese și care trebuie captate și reprezentate
- existența mai multor surse de cunoștințe
- dificultatea de a identifica, din masa de cunoștințe pe cele necesare sau relevante

- necesitatea de a colecta un număr cât mai mare de excepții și de cazuri nespecifice pentru a putea conferi suficientă flexibilitate viitorului sistem. Spre deosebire de sistemele tradiționale, un sistem expert își propune să rezolve probleme nestructurate în care excepțiile sunt frecvente.

Se poate identifica o latură statică, descriptivă a cunoștințelor și o latură dinamică, ce privește modul de participare a acestora în efectuarea raționamentelor. Această diferențiere se observă și în colectarea cunoștințelor, fiind de preferat să se obțină o descriere cât mai complexă înaintea abordării modului în care expertul le corelează și le utilizează în rezolvarea problemelor.

Achiziția cunoștințelor trebuie să asigure:

- identificarea obiectelor -acest proces nu trebuie înțeles numai ca o enumerare ci ca un proces de atribuire de identificatori unici și de rezolvarea problemelor legate de eventualele ambiguități, sinonime și omonime
- descrierea obiectelor, conceptelor sau a noțiunilor folosite -se face prin precizarea atributelor și a valorilor relevante ale acestora de multe ori prin exprimări pronunțat calitative precum “bun”, “mediocru”, “opus” etc.
- organizarea (structurarea) cunoștințelor, care constă în identificarea și precizarea rețelilor ierarhice ale obiectelor, conceptelor sau noțiunilor folosite de către expert.

Odată acest cadru definit, se poate trece la identificarea și explicarea regulilor și strategiilor de raționament aplicate de expertul uman. În virtutea caracterului iterativ al etapei, după o primă definire a cunoștințelor, care practic nu poate fi exhaustivă, se trece la abordarea strategiilor și proceselor de raționament a căror analiză antrenează revederea și completarea cadrului conceptual.

Achiziția cunoștințelor se poate face pe mai multe căi⁸⁰:

-manuale –în general se face prin interviu, utilizat frecvent și în realizarea aplicațiilor convenționale. Metodele manuale sunt lente, scumpe și uneori lipsite de acuratețe. Fluxul de prelucrări în cadrul acestei metode se poate reprezenta grafic astfel (figura nr. 24):

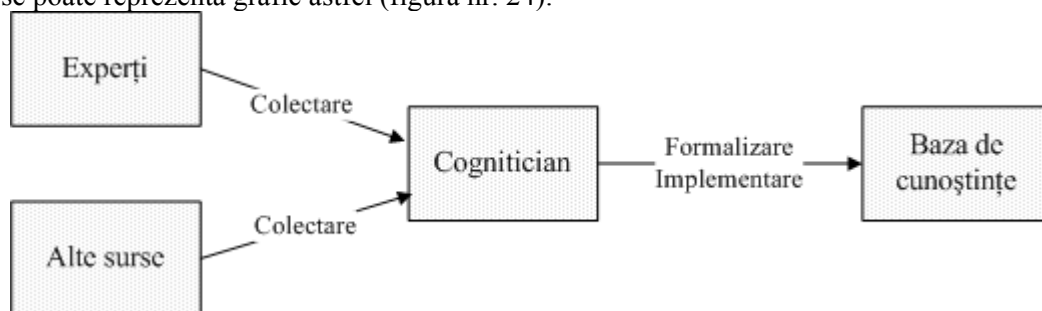


Fig. 24 Fluxul de prelucrări în cadrul metodelor manuale de achiziție a cunoștințelor

Datorită nivelului de incertitudine ridicat, s-au definit diverse tehnici alternative, cum ar fi grilele de repertoriere. După identificarea cazurilor de rezolvat și a caracteristicilor pe care se bazează expertul în expertiza pe care o desfășoară, se întocmește această grilă, elementele analizate primind valori în funcție de extremele din stânga sau din dreapta grilei. Grila se umple cu valori de la 10 la 1, iar completarea grilelor de persoane diferite va permite compararea cunoștințelor diverșilor experți în domeniul respectiv.

-semiautomate –acestea, la rândul lor, se pot împărți în două categorii:

- a)cele care sprijină experții în construirea bazei de cunoștințe, fără aportul cogniticienilor
- b)cele care sprijină cogniticienii în executarea fazelor specifice achiziției cunoașterii, rapid și eficient, uneori cu puțin ajutor din partea experților. Fluxul de prelucrări și legăturile dintre componentele acestei metode se pot reprezenta astfel (figura nr. 25):

⁸⁰ [Andone99] Andone I. Țugui A. –Sisteme inteligente în management, contabilitate, finanțe-bănci, Editura Economică, București, 1999.

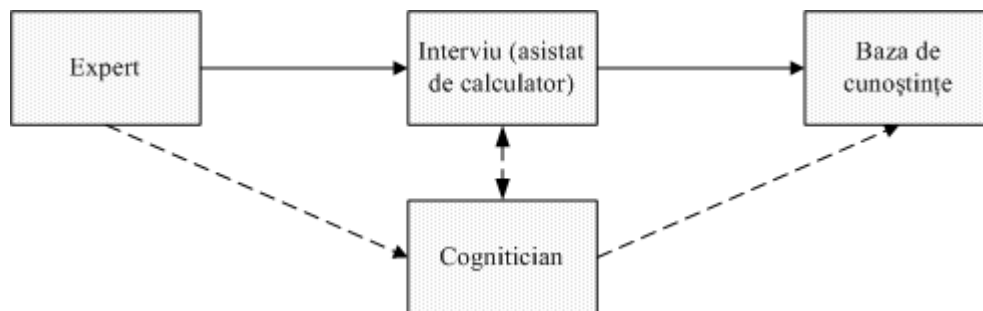


Fig. 25 Fluxul de prelucrări în cadrul metodelor semiautomate de achiziție a cunoștințelor

-*automate* -permit generarea automată a bazei de cunoștințe pe baza unor seturi de exemple sau cazuri formulate de expert. Rolul cogniticianului și expertului este minimizat sau chiar eliminat. Cea mai cunoscută metodă este *metoda inducției* prin care achiziția cunoașterii o poate efectua chiar analistul proiectant, rolul expertului fiind minim, iar de cognitician nici nu este nevoie. În cazul acestei metode, reprezentarea fluxului de prelucrări și a legăturilor dintre componentele metodei este următoarea (figura nr. 26):

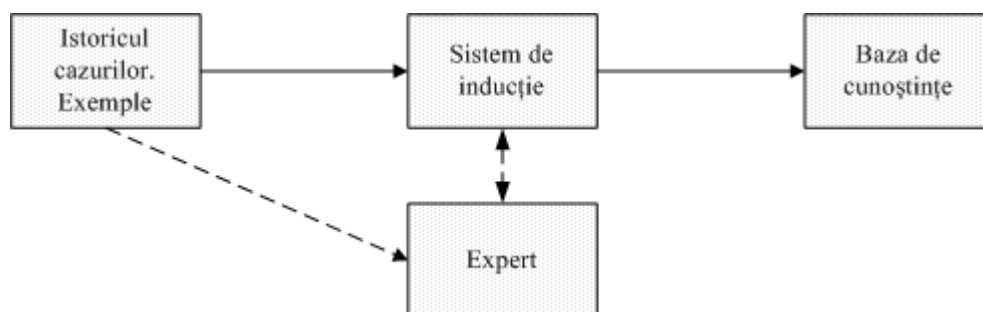


Fig. 26 Fluxul de prelucrări în cadrul achiziției automate a cunoștințelor prin metoda inducției

Achiziția cunoștințelor se poate face utilizând două tipuri de abordări:

- *metoda studiilor de caz (a eșantionării)* –aceasta se bazează pe concentrarea activității pe un singur eșantion de probleme. Cogniticianul captează explicațiile expertului folosind una din formele de reprezentare a cunoștințelor, explicații ce vizează eșantionul ales. Se continuă apoi cu celelalte eșantioane până se ajunge la baza de cunoștințe finală. Pe baza eșantioanelor studiate se pot face apoi generalizări.
- *metoda globală* –aceasta se bazează pe o abordare sistemică și are în vedere un sistem de idei și tehnici aplicabile la un număr mare de situații. Întrebările cogniticianului vor fi astfel înlănțuite încât să surprindă toate aspectele cunoașterii din acel domeniu folosind această perspectivă.

4.3.1.10. Reprezentarea cunoașterii

Problema reprezentării cunoștințelor din domeniul de expertiză considerat stă în centrul preocupărilor proiectanților de sisteme expert, această reprezentare determinând dezvoltarea sistemului și interclasându-se cu strategiile de control. Pot fi utilizate mai multe maniere de reprezentare pentru aceleași cunoștințe, alegerea modalității potrivite făcându-se în funcție de natura problemei, de instrumentația disponibilă, de oportunitate, de intuiție etc., reprezentarea căzând în sarcina inginerului de cunoștințe.

Prima fază din activitatea inginerului de cunoștințe este achiziția cunoștințelor. Pe parcursul acestei achiziții se identifică conceptele fundamentale ale domeniului de expertiză, se asimilează regulile, strategiile și raționamentele expertului uman. Următorul pas este structurarea acestor cunoștințe după care se poate alege un sistem de reprezentare a cunoștințelor. Acesta va depinde de regulă în cea mai mare măsură de sistemul disponibil atunci când cel care elaborează sistemul expert va dispune de software-ul necesar. Dacă nu există sistem disponibil, alegerea va depinde de termenul de realizare impus și de resursele financiare ale celui care a comandat sistemul. Cunoștințele pot fi adaptate -mai mult sau mai

puțin ușor- și unor șabloane deja existente. Cu cât șablonul este mai potrivit, cu atât vor fi mai convingătoare performanțele sistemului, iar întreținerea mai facilă.

Faptele sunt reprezentate de reguli (din punct de vedere informatic) sub forma unor șiruri structurate de caractere. Un procedeu răspândit este folosirea unor obiecte cu valoare atribuită de genul:

$\langle \text{obiect} \rangle \langle \text{atribut} \rangle \langle \text{valoare} \rangle$

unde atributul este o proprietate a obiectului. Se poate utiliza și sistemul echivalent în esență, dar mai simplu (care permite și folosirea unor variabile ca și operator) astfel:

$\langle \text{obiect} \rangle \langle \text{operator} \rangle \langle \text{valoare} \rangle$

Predicatele sunt folosite de regulă pentru exprimarea relațiilor dintre obiecte. Considerăm un exemplu imaginar: pentru a exprima faptul că firma X deține 20 de acțiuni din capitalul societății ALFA, putem exprima aceasta sub forma unei funcții cu două variabile:

$\text{Parte capital}(X, ALFA) = 20$

sau folosind forma obiectelor cu valoare atribuită:

$\langle X \text{ posedă} \rangle \langle ALFA \rangle \langle 20 \rangle$.

O funcție de n variabile se poate scrie oricând sub forma unui predicat de ordinul $n+1$, unde argumentul al $n+1$ -lea va fi valoarea funcției. Deci relația exprimată înainte printr-o funcție cu două argumente se poate scrie sub forma unui predicat de ordinul 3 astfel:

$\text{Structură capital}(X, ALFA, 20)$.

Reducând această expresie sub forma unor predicate de ordinul 2 se obține:

$\text{Posedă}(X, \$1)$

$\text{Este posedat}(ALFA, \$1)$

$\text{Unități}(20, \$1)$;

unde am introdus un obiect fictiv (\$1) care poate fi interpretat ca un act pentru care factorii implicați sunt detaiați cu ajutorul predicatelor de ordinul 2. O funcție se poate restrânge deci la predicat, iar predicatele de ordinul 2 pot exprima predicate de ordin superior. Aceasta explică faptul că majoritatea sistemelor existente se restrâng la predicate de ordinul 2 și la relații binare.

Pentru a regrupa în mod adecvat numeroase informații legate între ele, se pot utiliza ceea ce Minsky a numit “*frame*”-uri. “*Un frame (obiect structurat) este o structură de date pentru reprezentarea unei situații stereotip. Fiecărui frame îi sunt atașate diverse tipuri de informații. Unele din acestea se referă la modul de utilizare al frame-ului. Altele se referă la ce poate cineva să aștepte să se întâmple, iar altele se referă la ce trebuie făcut atunci când așteptările nu se confirmă*”⁸¹.

Un “*frame*” (cadru) este o unitate de informație care regroupează un număr de “*sloturi*” (rubrici). Frame-urile asigură organizarea cunoștințelor în ierarhii sau rețele succesoriale, prin care are loc partajarea proprietăților comune. Din punct de vedere al structurii, un **frame** se compune din⁸²:

-*nume* –servește identificării frame-ului în baza de cunoștințe

-*antecedenți* –permit definirea ierarhiilor cu transfer succesoral al proprietăților. În afara nivelului cel mai ridicat, orice cadru trebui să aibă cel puțin un antecedent.

-*rubrici (slots)* -sunt elemente care stochează proprietățile frame-ului.

O **rubrică** se compune din:

-*nume* –servește pentru identificare și trebuie să fie unic numai în interiorul frame-ului în care apare

-*valoare asociată* -fiecare rubrică are atașate proceduri care gestionează valoarea rubricii. Ea poate fi vidă în momentul definirii rubricii și poate face obiectul operațiilor de atribuire și consultare.

-*predicatele asociate* (dacă există)

Un cadru moștenește rubricile tuturor antecedentilor (părinților) săi. Este permisă existența de cadre fără rubrici, astfel putându-se asigura flexibilitatea necesară în reprezentarea unor concepte abstracte.

Pentru a asigura caracteristica de incertitudine a valorilor, în cadrul sloturilor se pot introduce și defini valori fuzzy sau se pot introduce valori aproximative sau implicite.

În reprezentarea cunoștințelor prin cadre noțiunea propriu-zisă de regulă dispare. În timpul sesiunii, rețeaua de cadre este parcursă (și modificată dacă este cazul) din aproape în aproape cu ajutorul procedurilor atașate rubricilor. Această cvasi-dispariție a regulilor nu contribuie la transparența bazei de

⁸¹ [Minsky74] Minsky M. -A Framework for Representing Knowledge -MIT AI Laboratory Memo, 306, June 1974.

⁸² [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

cunoștințe și nu ușurează scrierea sistemului. De aceea, sisteme bazate pe cadre reintroduc adesea regulile (sub forma unor cadre particulare). De multe ori se întâmplă să coexiste regulile “clasice” și cadrele într-un sistem expert.

Conceptul de frame se deosebește de cel de structură a unei înregistrări (întâlnit ca structură de date în diverse limbaje de programare) prin două caracteristici: moștenire (transferul succesoral al proprietăților prin care fiecare element moștenește proprietățile elementelor aflate pe nivelul superior) și predicatelor asociate.

Considerăm că avantajele reprezentării prin cadre sunt:

- organizarea și structurarea unei rețele relaționale prin dependențele dintre fapte precum și partiționarea (sau compartimentarea) acestor fapte – operație utilă în faza de restricționare sau de filtrare. Utilitatea clasificărilor apare mai ales la organizarea regulilor sau la introducerea metaregulilor.
- asigurarea valori implicite adică valorile asociate unei rubrici atunci când utilizatorul nu are informații precise pentru a indica o valoare.
- reprezentarea și tratarea excepțiilor
- transferul succesoral asigurat prin moștenire crează premisele menținerii coerenței logice în baza de cunoștințe Astfel fiecare cadru “știe” care sunt valorile dependente situate în alte cadre care trebuie schimbate atunci când conținutul unei rubrici se modifică. Modificarea unei rubrici pe un anumit nivel duce la propagarea automată pe toate nivelurile succesive a valorilor dependente, fără nici o intervenție din afară.

Dezavantajul cadrelor constă în faptul că nu sunt foarte bune pentru a descrie structurile de legături dintre obiecte. De aceea pentru descrierea legăturilor se folosesc scenariile. Scenariile sunt definite ca structuri care descriu secvențe de evenimente care se petrec stereotip, adică se petrec normal într-un context dat. Scenariile permit părăsirea anumitor detalii care pot fi completate cu ajutorul cunoștințelor pe care le au cei ce recepționează informația. Ca formă, un scenariu face să se înlăture o serie de sloturi care nu au fost completate. Această structură este o construcție interdependentă care permite navigarea de la un slot la altul.

Frazele limbajului natural (subiect, atribut, verb, complement) se pretează bine reprezentării prin intermediul *rețelelor semantice*. Primul model de reprezentare a cunoașterii cu ajutorul rețelelor a fost propus în 1966 de M. Ross Quillian și a fost denumit rețea semantică deoarece avea ca obiectiv captarea cu ajutorul acestui formalism a părții obiective a semnificației pe care oamenii o dau cuvintelor limbajului natural, pentru a putea utiliza această reprezentare la efectuarea de raționamente care ar presupune înțelegerea acestor semnificații. Noțiunea de rețea sau altfel spus *graf de obiecte* provine deci din analiza limbajului, de unde și denumirea de *rețea semantică* (legată de înțelesul frazelor) atribuită unei asemenea rețele din obiecte. În acest caz, nodurile rețelei sunt asociate cuvintelor (ce descriu concepte), iar arcele reprezintă relațiile cu alte cuvinte. Schank a arătat că aceste reprezentări grafice pot fi adaptate la analiza expunerilor, povestirilor etc. Aceste reprezentări sunt utilizate de mai mult timp de către analiști (pentru analiza programelor), tehnologia sistemelor expert doar reluând ideea pentru structurarea cunoștințelor de specialitate. Reprezentarea cunoașterii prin rețele semantice este bazată pe utilizarea grafelor pentru a descrie relații între concepte, fapte, evenimente etc. O rețea semantică este un graf orientat în care atât nodurile cât și arcele sunt etichetate. În mod obișnuit nodurile reprezintă obiecte, noțiuni sau situații din domeniul specific iar arcele reprezintă relații. O bună structurare a rețelei semantice evită pe cât posibil circuitele în graf, tinzând spre o organizare ierarhică a obiectelor. O astfel de organizare se traduce printr-o arborescență de fapte sau de fapte primordiale (elementare) ceea ce ușurează structurarea domeniului. O arborescență similară pentru anumite obiecte importante este utilizată în MYCIN și EMYCIN sub numele de *arbore de contexte* (de exemplu contextului Pacient i se va substitui numele bolnavului).

Mergând pe această cale, se poate realiza o arborescență a clasificării sau *taxonomie* (similară cu cele existente în botanică sau zoologie). O taxonomie este o arborescență în care există proprietăți de *moștenire*. Clasificarea se face în general de la general la particular (de exemplu: Acțiunile “Terapia” fac parte din categoria “Acțiuni” care sunt “Valori mobiliare”). Noțiunea de taxonomie se utilizează în unele cazuri și pentru grafuri ierarhizate –care nu sunt neapărat structuri arborescente- în care există proprietăți de moștenire (dar pot exista mai multe moduri de a ajunge la același obiect, mai ales pentru nivelurile inferioare ale ierarhiei). O taxonomie bine concepută poate să reducă considerabil numărul caracteristicilor atașate în mod explicit unui obiect.

De fapt, orice arborescență din reprezentarea faptelor sau regulilor poate fi folosită de către mașina inferențială. Chiar dacă arborescența se aplică inițial doar la fapte, structurarea aleasă acționează în mod inevitabil și asupra regulilor (alternativele de reprezentare ale faptelor și regulilor sunt strâns legate). Rețelele celor mai multe sisteme bazate pe cadre sunt organizate sub formă ierarhică cu moștenire.

Este important de menționat că toate aceste sisteme de reprezentare a conceptelor (obiectelor) sunt inspirate mai mult sau mai puțin din caracteristicile umane. Acest fapt este confirmat și de psihologia cognitivă, care are ca obiect de studiu reprezentarea cunoștințelor la om. Astfel, existența unităților de cunoaștere care comportă mai multe elemente, fie la nivel descriptiv fie la nivel operațional, este atestată în literatura de specialitate. Majoritatea acestor noțiuni (de exemplu: model, obiect, plan, prototip, scenariu, schemă, script etc) sunt destul de asemănătoare cu cea a cadrului. Este adevărat că noțiunea de prototip în psihologia cognitivă desemnează un obiect reprezentativ al unei categorii. În această accepțiune, celelalte obiecte ale categoriei sunt instanțieri diferite ale prototipului (sau ale cadrului tipic).

Multitudinea de metode de reprezentare a cunoștințelor provine și din necesitatea de a acoperi un spectru larg de domenii de utilizare a sistemelor expert. Uneori, se folosesc chiar soluții combinate, în care alături de cadre coexistă reguli explicite, reguli ce pot fi invocate de către predicatele asociate și care, la rândul lor, să poată folosi în calitate de fapte conținutul cadrelor. Convergența cu sistemele de baze de date recomandă utilizarea cadrelor și a altor metode de reprezentare asemănătoare, în special cele orientate obiect, ca soluții fundamentale în realizarea bazelor de date inteligente⁸³. Rețelele semantice sunt utilizate și pentru reprezentarea cunoștințelor din domeniul financiar, existând sisteme expert comercializabile pentru analiza financiară în care reprezentarea cunoștințelor se face prin intermediul rețelilor semantice.

4.3.1.11. Reprezentarea regulilor

Nu întotdeauna este evidentă diferența dintre fapte și reguli, distincția acestora depinzând în mare măsură de reprezentarea adoptată. Problema esențială a sistemelor bazate pe reguli este modul de structurare al ansamblului de reguli pentru obținerea unei eficacități maxime în procesele de filtrare și control. Atunci când există o structurare arborescentă a domeniului (sau o taxonomie) se poate utiliza aceasta pentru a construi o *rețea de reguli* (sau o *rețea de inferențe*, un *graf de precedență a regulilor*) care indică ordinea de aplicare a regulilor. Rețelele de inferență furnizează o reprezentare grafică agreabilă care ajută expertul să-și vizualizeze structura cunoașterii pentru a urmări raționamentele.

Nu este suficientă numai exprimarea cunoașterii brute sub forma unui set de reguli pentru ca sistemul să treacă la raționamente. Este necesar să adăugăm și comenzi prin care se precizează modul în care se utilizează și structurează cunoașterea. Este vorba aici despre cunoașterea numită de obicei metacunoaștere. Există și reguli care guvernează metacunoașterea. Aceste reguli, care reprezintă practic niște cunoștințe despre reguli, se numesc *metareguli*. O metaregulă este scrisă în general ca o regulă "obișnuită". De obicei gestiunea metaregulilor se bazează pe o mașină inferențială specială. Seturile de reguli care sunt dirijate de metareguli sunt în mod obișnuit definite de către expert (ele fac parte din baza de cunoștințe). De fapt, toate organizările de reguli și/sau de obiecte reprezintă metacunoștințe. Această deosebire între nivelurile de cunoștințe a condus la redescoperirea noțiunilor de niveluri de reprezentare. Acestea sunt dificil de fixat și de inventariat pentru că depind în mare măsură de domeniul ales și de problema pusă.

Există și metode de reprezentare destinate în special reprezentării cunoștințelor la nivelul bunului simț, cunoștințelor vagi sau excepțiilor. Dacă ne interesează rezolvarea unei probleme reale, experiența a demonstrat că metodele amintite pot fi satisfăcătoare și suficiente. Multe din problemele teoretice, precum excepțiile sau non-monotonia, au fost rezolvate sau pot fi ocolite de către expert. De exemplu, în majoritatea proceselor temporale non-monotonia nu reprezintă o problemă dacă mecanismul de inferență acționează mai rapid decât timpul necesar modificării fizice semnificative a sistemului. Problemele de excepție pot fi stăpânite cu succes de către expert, noțiunea de excepție neavând de altfel sens decât în raport cu contextul (excepția depinde de funcția de utilitate a celui care o consideră). Alegerea metodei de

⁸³ [Lițoiu98] Lițoiu V, Bădică A., Tărbujaru V. –Sisteme expert financiar-bancare –curs, Reprografia Universității din Craiova, 1998

reprezentare a cunoștințelor depinde de criterii ca: transparența (lizibilitate) bazei de cunoștințe, facilitatea ingineriei cunoașterii, disponibilitatea și costul unui instrument de elaborare. De cele mai multe ori aceasta este o problemă de obișnuință și de oportunitate.

4.3.1.12. Mașina de inferență

Exemplul anterior (cu umbrela) este util, dacă plouă, dar un domeniu de cunoaștere alcătuit doar din câteva reguli nu necesită un sistem expert. Dacă însă numărul regulilor crește la zeci, sute sau mii, atunci crește și complexitatea nu numai datorită numărului de reguli ci și din cauza relațiilor dintre reguli.

Extinzând puțin exemplul anterior al umbrelei (pentru ilustrarea deducției înainte și înapoi) putem stabili câteva reguli:

- R1:** Dacă plouă,
Atunci luați umbrela.
- R2:** Dacă cerul este înnorat și trotuarul este ud,
Atunci plouă.
- R3:** Dacă cerul este gri,
Atunci cerul este înnorat.
- R4:** Dacă luați umbrela,
Atunci puteți uita umbrela în autobus.
- R5:** Dacă lăsați urme,
Atunci trotuarul este ud.

Mașina de inferență dă sens acestor reguli prin legarea lor, pentru a putea adresa utilizatorului întrebările corespunzătoare și pentru a putea realiza concluziile corespunzătoare. Setul prezentat ne poate ajuta să înțelegem dacă avem sau nu nevoie de umbrelă. În acest caz chestionarea ar începe prin aplicarea întrebării cuprinse în regula R3: Cerul este gri? care ar activa regula R5 Lăsați urme? urmând activarea succesivă a regulilor R2, R1, R4. Consultarea regulilor ne dă rezultatul: Uitați umbrela în autobus (conform strategiei de urmărire înainte). Dacă am dori să întrebăm ce condiții au dus la uitarea umbrelei în tren, mașina de inferență va parcurge drumul înapoi de la regula R4 la R1, apoi la R2, R3, R5 și la întrebarea: De ce am uitat umbrela în autobus? va răspunde: Pentru că lăsați urme și cerul este gri (conform strategiei de urmărire înapoi). Acest exemplu ilustrează diferența dintre deducția prin înlănțuire înainte (*forward chaining*) și deducția prin înlănțuire înapoi (*backward chaining*).

Un alt exemplu de raționare înainte și raționare înapoi poate fi ilustrat astfel:

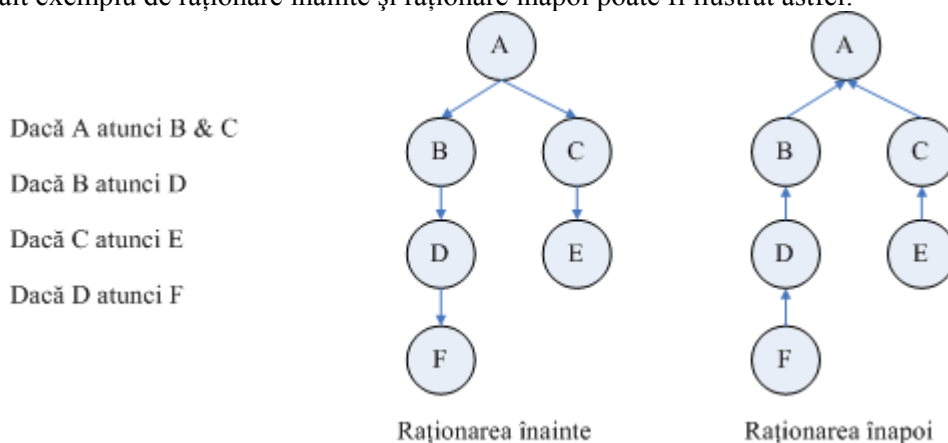


Fig 27 Raționarea înainte și raționarea înapoi

Dacă numărul răspunsurilor posibile este cunoscut și redus, atunci deducția prin înlănțuire înapoi este foarte eficientă. Dacă însă, numărul răspunsurilor posibile este foarte mare, atunci se recomandă deducția prin înlănțuire înainte. Cele mai multe sisteme expert înclină spre folosirea inferenței sub forma deducției înlănțuite înapoi deoarece rezultatul este cunoscut și este nevoie de factorii care conduc la acel

rezultat (cazul clasic al sistemelor de diagnosticare: ne interesează, de exemplu, aflarea factorilor care conduc la apariția cancerului pulmonar).

Baza de cunoștințe dispune de două dimensiuni. Această caracteristică determină capacitatea sistemului de a efectua pași nu numai înainte (prin înlănțuirea înainte) și înapoi (prin înlănțuirea înapoi) ci și lateral, adică să consulte regulile aflate la același nivel în baza de cunoștințe. Dacă sistemul este foarte mare, atunci utilizatorul nu poate efectua căutări complexe în baza de cunoștințe, din cauza limitărilor de timp sau a performanțelor calculatorului. Aceasta înseamnă o căutare relativ restrânsă, deci nu se verifică întotdeauna toate posibilitățile, din cauza limitărilor de timp sau a performanțelor calculatorului. Este foarte posibil ca mașina de inferență să execute căutarea întâi în adâncime. În acest caz se începe cu întrebarea de deschidere după care se avansează, prin reguli urmărind înlănțuirea lor înainte, până la obținerea unui răspuns. Acest răspuns, bineînțeles, se poate întâmpla să nu fie cel mai bun. Deoarece dacă mașina de inferență ar fi parcurs o altă serie de întrebări (și reguli) am fi putut obține un alt răspuns. Cele mai multe programe funcționează după acest model, dar în sistemele mai rafinate mașina inferențială permite utilizatorului să aleagă modalitatea chestionării.

Pentru a ilustra diferența dintre cele două abordări, putem să ne imaginăm un specialist, față de un om care posedă doar cunoștințe generale. Primul va pune întrebări din ce în ce mai detaliate până se convinge asupra rezultatului. Cel de al doilea va pune întrebări la întâmplare, într-un mod aparent nepotrivit, până la rezolvarea într-un fel sau altul a problemei.

În cadrul unui sistem mic, unde există posibilitatea căutării fiecărei reguli, procedeul de întrerupere a căutării nu este necesar, deși procedeele destinate scurtării timpului de căutare nu se pot ignora. Unul din aceste procedee (care permit utilizatorului experimentat reducerea timpului de căutare) este furnizarea fără cerere expresă a unor informații sistemului. Aceste informații pot preîntâmpina folosirea unor grupuri mari de reguli. De exemplu, în cadrul unui sistem pentru planificarea concediului de vacanță, putem evita toate întrebările referitoare la condițiile meteorologice dacă precizăm de la început că acestea ne sunt indiferente.

4.3.1.13. Mecanismele de inferență ale unui sistem expert

Un sistem expert poate fi privit și ca un sistem de producții capabil să reproducă raționamentele unui expert într-un domeniu bine definit (eventual chiar în mai multe domenii bine definite). Ce face un expert? El pleacă de la o situație descrisă de un număr de fapte sau de date pe care le analizează și pentru care dă verdicte. Aceste verdicte nu trebuie înțelese ca soluții, ele sunt rezultate firești ale raționamentelor expertului uman. Deci, plecând de la fapte, experții elaborează înlănțuiri de raționamente, raționamente care pot fi reprezentate sub forma unor reguli de producție de tipul:

Dacă condiția inițială este satisfăcută, atunci execută acțiunea.

Varietatea contextelor acoperite de acest formalism este foarte largă. Printre regulile de acest gen, un loc aparte este ocupat de *regulile deductive* (inferențe), care sunt de tipul:

Dacă premiza este adevărată, atunci concluzia este adevărată.

Regula de producție comportă una sau mai multe premise (condiții) care conduc la una sau mai multe concluzii (acțiuni). Premisele conțin faptele, iar concluziile definesc cuantele de cunoaștere propriu-zisă.

Regulile de producție de acest tip se află la baza majorității sistemelor expert. Ele fac parte din tipul general definit mai sus, în sensul în care condiția inițială este satisfăcută dacă premiza se găsește în baza de fapte adevărate (excepția regulii conduce la adăugarea concluziei în aceeași bază de fapte adevărate). Acest tip de logică deductivă a fost folosit și de greci în antichitate și aparține de domeniul logicii. Regulile gramaticale dintr-un limbaj indică cum pot fi construite corect frazele, plecând de la cuvinte. Regulile de producție care se întâlnesc în cadrul sistemelor expert sunt reguli de acțiune (ele permit fie să se acționeze, fie să se îmbogățească informația înainte de a se acționa).

Principalul dezavantaj al regulilor de producție este legat de decuparea cunoașterii într-un mod puțin natural. Regulile de producție nu pot exprima în totalitate modul în care un expert își organizează

cunoașterea. Ele rămân la nivelul începătorului care aplică reguli elementare de acțiune. În ciuda acestui dezavantaj, reprezentarea cunoașterii prin reguli de producție reprezintă cea mai utilizată metodă.

Ciclul de bază al unui motor de inferență comportă cinci etape: restricționarea, filtrarea, rezolvarea conflictelor, selecția și execuția propriu-zisă (declanșarea regulilor alese). Fluxul de înlănțuire a celor cinci etape și legăturile dintre ele este reprezentat grafic în figura 28.

Într-un sistem expert, este necesar să existe o stare a bazei de fapte prin care să poată fi recunoscute regulile aplicabile, operație care este numită *filtrare* (pattern matching). Filtrarea este o operație destul de lungă, iar sistemele recurg în general la preselecția regulilor care urmează să fie filtrate. În loc să fie selectate regulile aplicabile din mulțimea tuturor regulilor, se consideră o submulțime de reguli din care se selectează seturi aplicabile. Această operație de preselecție dinaintea filtrării se numește *restricționare*. După ce regulile aplicabile au fost recunoscute, trebuie aleasă o strategie de control (operație definită ca *selectare*). După ce regula a fost selectată, ea trebuie să fie și executată. Modulul central al sistemului expert care realizează aceste operații (restricționarea, filtrarea, selectarea și executarea) se numește *motor de inferență* (*mecanism, mașină* etc). sau *interpretor*.

Filtrarea constă în determinarea regulilor aplicabile potențial pentru o stare dată a bazei de cunoștințe. Sistemul de control poate decide ca o parte din reguli să nu fie testate la acest nivel (prin restricționare). Partea care trebuie testată dintr-o regulă se numește *declanșator*. Acesta este de regulă o formulă, alcătuită corespunzător în sensul logicii propozițiilor sau logicii de ordinul întâi, care se referă la fapte oarecare (conținând fapte și conectori). Sintaxa acestor formule este standardizată, iar numărul conectorilor acceptați este în mod obișnuit limitat.

Din alt punct de vedere, filtrarea este o operație greoaie în sistemele bazate pe logica predicatelor de ordinul întâi, care poate conduce la o explozie combinatorică a numărului de teste. Pentru a ocoli acest impediment, se poate aplica o metodă “naturală” care constă din reducerea din amonte a numărului de reguli filtrate, prin aplicarea procedurii de restricție. La nivelul filtrării se poate ameliora procesul examinat (referitor la declanșatorul fiecărei reguli) dând prioritate predicatelor care au mai puține variabile libere. Un alt mod de a proceda este păstrarea în memorie a rezultatelor filtrării precedente pentru fiecare regulă sau pentru fiecare parte a declanșatorului (de exemplu se pot reține unificările posibile pentru fiecare parte a declanșatorului).

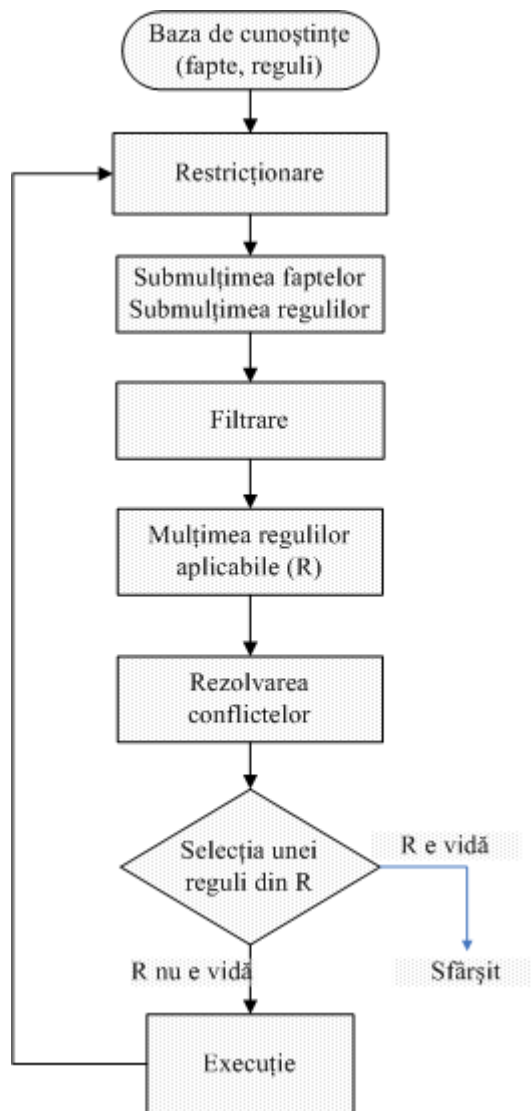


Fig. 28. Etapele ciclului de bază al unui motor de inferențe

După fiecare ciclu al motorului de inferență se actualizează conținutul acestei zone de memorie. Această metodă de filtrare este denumită în literatura de specialitate *filtrare prin propagare*. Greutățile în cadrul operațiilor de filtrare apar mai ales în timpul execuției (când motorul de inferență este solicitat de către utilizator). De cele mai multe ori se reduce timpul de execuție prin realizarea unei precompilări/indexări a bazei de reguli (înainte de folosirea motorului inferențial). Această compilare prealabilă crează o rețea protejată (prin pointere) de interdependențe între reguli. Principiul acestei rețele constă în crearea unor legături care, de exemplu, precizează dacă o regulă a cărei concluzie conține “evaluarea proiectului” să nu fie filtrată atâta timp cât “timpul de revenire” nu a fost evaluat, stabilindu-se astfel ordinea de aplicare a setului de reguli. În timpul execuției, procesul poate fi accelerat dacă propagarea faptelor adevărate se efectuează pe baza acestei rețele pregătite. Această metodă de precompilare a regulilor a fost folosită pentru prima dată pentru ameliorarea performanțelor sistemului OPS 5 (sub denumirea “algoritmul RETE”). În cazul prezentat mai sus, rețeaua creată rămâne invizibilă, ascunsă pentru utilizatori.

Dacă, după faza de filtrare, există un număr de reguli cunoscute ca fiind aplicabile, structura de control va fi cea în măsură a stabili care dintre ele să se aplice (execute). Concluzia regulii (sau premiza în cazul sistemelor guvernate de reguli) are ca efect modificarea bazei de cunoștințe prin crearea, modificarea sau ștergerea unor înregistrări (fapte). Bineînțeles, se pot imagina și sisteme în care concluzia unei reguli să modifice mulțimea regulilor pe care nu le-am analizat, însă un asemenea sistem nu ar fi un

sistem de producții în accepțiunea clasică a definiției. Mai multe sisteme expert se rezumă, sub acest aspect, doar la adăugarea de fapte în baza de fapte. Aceste sisteme sunt denumite *monotone*. Pe drept cuvânt, denumirea de monotonică presupune că faptele care se adaugă nu sunt contradictorii cu cele care se găsesc deja în baza de fapte. Altfel spus, aceste sisteme nu țin seama de loialitatea unui fapt cunoscut în prealabil ca adevărat. Această limită este de altfel un obstacol în realizarea bazelor de cunoștințe relative, la expertizele variabile în timp. Uneori se folosesc reguli ale căror concluzii nu sunt adevărate decât cu un grad de certitudine. Această situație se întâlnește adesea în sistemele expert destinate diagnosticării.

În general se poate afirma că, într-un sistem de producții, strategiile de control permit trecerea unei baze de fapte într-alta (sistemul trece dintr-o stare în alta). Fiecărei stări a bazei de fapte îi corespunde o submulțime de reguli de producție bine determinate (*reguli aplicabile*). Pentru a reduce neajunsul referitor la numărul de teste de efectuat, se poate lua decizia ca numai o parte din reguli sau fapte să fie testate. Deci înaintea operației de filtrare este necesară o fază de restricționare. Sistemul va funcționa, în acest caz, conform schemei reprezentate în figura nr. 28. Modulul de selecție trebuie să dispună de capacitate de acțiune și în cazul în care mulțimea regulilor aplicabile este vidă (dacă restricția este prea puternică, se poate încerca din nou cu alta mai puțin restrictivă). După execuție se poate relua o regulă din mulțimea R, până la epuizarea tuturor elementelor din mulțime sau se poate reface întreg ciclul. Trebuie să facem distincție între cele două faze de alegere care intervin: faza de alegere dinaintea filtrării (*restricția*) și faza de alegere de după filtrare (*selecția*). Faza de selecție se poate numi și *rezolvarea conflictului*, în acest caz mulțimea regulilor aplicabile la acest nivel numindu-se *mulțimea conflictelor*. Strategiile de control aplicabile acestor două faze pot fi diferite.

Pentru a reduce mulțimea de fapte sau de reguli prin operația de filtrare, au fost propuse mai multe modalități. Într-un sistem guvernat de fapte, restricționarea constă din luarea în considerare a faptelor adevărate. Filtrarea constă din reținerea tuturor regulilor a căror premiză este adevărată, ținând seama de faptele (adevărate) prezentate în baza de fapte. Se spune că un astfel de sistem funcționează prin înlănțuire înainte (*forward chaining* sau *raționament deductiv*). Dacă sistemul este guvernat de scopuri, restricția constă din luarea în considerare doar a scopurilor (acestea pot fi privite și ca fapte ce trebuie stabilite). Filtrarea constă deci din reținerea regulilor a căror concluzii conțin scopuri care se găsesc în baza de fapte. Se spune că un astfel de sistem funcționează prin înlănțuire înapoi (*backward chaining* sau *raționament inductiv*). Dacă sistemul este guvernat de fapte și scopuri se poate spune că sistemul are înlănțuire mixtă (*mixed chaining* sau *raționament mixt*).

Raționamentul deductiv – în acest caz, motorul de inferențe pleacă de la fapte și caută un anumit scop (obiectiv). Raționamentul deductiv aduce două mari avantaje, care în unele aplicații sunt absolut necesare:

- permite posibilitatea generării tuturor soluțiilor posibile
- îmbogățește la fiecare ciclu al motorului de inferențe baza de fapte, ceea ce simplifică deducțiile ulterioare.

Totuși acest raționament considerăm că prezintă și unele inconveniente, cum ar fi:

- baza de fapte nu conține suficiente fapte inițiale pentru ca sistemul să poată atinge scopul urmărit
- sunt declanșate toate regulile aplicabile, chiar dacă unele nu prezintă interes
- în unele situații, când numărul de reguli și de fapte este foarte mare iar scopul nu poate fi atins, există pericolul ajungerii la o explozie combinatorie privind numărul de cicluri de bază ale motorului de inferențe
- nu este interactiv în timpul inferențelor, ceea ce face ca în situațiile de eșec în care numărul de fapte necunoscute se reduce la unul singur (sau un număr mic), utilizatorul să nu poată cunoaște această situație, în care ar putea interveni.

Raționamentul inductiv pleacă de la un scop (o problemă de rezolvat) pe care îl descompune în subprobleme până când obține probleme primitive (fapte dovedite sau interogabile). De aceea se spune că acest mod de raționament este dirijat de scop în sensul găsirii faptelor care permit atingerea acestuia (invers decât raționamentul deductiv). Regulile selecționate sunt acelea care au în partea de concluzii scopul specificat inițial. Premizele acestora vor deveni subscopuri ce urmează să fie demonstrate. Acest proces se repetă până când toate subscopurile obținute sunt demonstrate sau în urma etapei de filtraj mulțimea regulilor declanșabile este vidă (situație de eșec).

Avantajul principal al acestui mod de raționament constă în primul rând într-un spațiu de memorie internă redus, deoarece arborele de căutare este mic. În al doilea rând, timpul de găsim al

soluțiilor este foarte scurt, deoarece căutarea se face doar pentru subscopurile care au o șansă de verificare. Unul din riscurile raționamentului inductiv este acela al ciclării: pentru a demonstra A trebuie demonstrat B, pentru a demonstra B trebuie demonstrat A.

Raționamentul mixt încearcă să realizeze compromisuri între avantajele și dezavantajele celor două moduri de raționament de bază: deductiv și inductiv. Principiul general al acestui mod de raționament este următorul: se fixează un scop, se determină faptele deductibile, se aplică mai întâi raționamentul inductiv, care va solicita utilizatorul pentru specificarea valorilor unor fapte necunoscute dar interogabile și apoi se aplică raționamentul deductiv pentru a deduce tot ce este posibil.

Prioritatea între cele două moduri de raționament de bază (deductiv și inductiv) în cadrul raționamentului mixt depinde de natura problemei și de modul de reprezentare a cunoștințelor. În general, raționamentul mixt se recomandă atunci când problema de rezolvat se pretează la un raționament deductiv dar spațiul de căutare este larg. În acest caz, aplicând raționamentul inductiv se restrânge spațiul de căutare pentru raționamentul deductiv.

Restricția poate fi comandată și explicit, de către expert, cu ajutorul regulilor (este vorba în acest caz de *metareguli*, adică reguli aplicabile regulilor). O metaregulă poate indica starea următoare a bazei de cunoștințe prin setul regulilor considerat pentru filtrare. Metaregulele pot indica grupele de reguli prioritare sau pot defini o ordonarea submulțimii regulilor. Restricționarea poate respecta și legi sau reguli care nu sunt comandate prin metareguli ci care rezultă din structurarea problemei. În acest caz se spune că există o *lege de precedență*. Aceasta înseamnă că regulile pot fi organizate într-un *graf de precedență* care ilustrează ordinea de parcurgere (și implicit de aplicare) a regulilor.

Cunoștințele pot fi organizate și sub forma unei rețele analoage rețelelor semantice utilizate în analiza sintactică a limbajelor. Acest mod de organizare este eficient atunci când se poate constitui cu adevărat o organizare ierarhizată a cunoștințelor. Un alt mod de a comanda restricționarea poate fi prin organizarea regulilor în pachete sau scheme. Plecând de la această organizare, se poate constitui o structură arborescentă din reguli. Toate aceste metode depind însă, în primul rând, de felul reprezentării adoptate pentru cunoștințe.

După faza de restricționare urmează filtrarea, putând ajunge în situația în care sunt aplicabile mai multe reguli. Este necesară efectuarea unei alegeri din mulțimea acestor reguli. După ce regula găsită a fost executată, programul trebuie să știe unde să se întoarcă. În cazul în care există cel puțin o regulă aplicabilă (mulțimea rezultată din filtrare nu este vidă), există două strategii posibile. Prima este cea a explorării în *lățime înainte*, care constă din considerarea unei reguli aplicabile (de obicei prima disponibilă) care se execută. Altfel spus, se execută succesiv toate regulile aplicabile, înainte de a testa dacă condiția de oprire este satisfăcută (dacă nu, se trece la un nou ciclu al motorului de inferență). Acest tip de *explorare exhaustivă* este folosită și în sistemul MYCIN. O altă strategie este găsirea unei reguli aplicabile urmând o metodă euristică, după care se trece la execuție (dar execuția acestei reguli nu modifică baza de fapte). Apoi se verifică dacă condiția de oprire este satisfăcută. Dacă nu este satisfăcută, se trece la un nou ciclu al motorului de inferență. Cea de a doua strategie se numește explorare prin *adâncime înainte*. Alegerea la acest nivel se supune unor reguli, ca la faza de restricționare. Euristica nu se bazează neapărat pe o funcție de evaluare, ea poate fi dirijată prin diverse metode.

Dacă nici o regulă nu este aplicabilă (mulțimea rezultată prin filtrare este vidă), atunci majoritatea sistemelor adresează întrebări utilizatorului, iar dacă utilizatorul nu știe sau nu dorește să răspundă, sistemul se poate opri. Se poate lua deci în considerare faza de restricționare și fără schimbarea ciclului motorului inferențial.

4.3.1.14. Motorul de elaborare și învățarea

Modul normal de funcționare al sistemului este cel de *exploatare*. Pe măsura derulării sesiunii de lucru, motorul inferențial va solicita (prin intermediul interfeței sale) utilizatorului faptele necesare care pot fi utile pentru elaborarea raționamentului și completarea bazei de cunoștințe inițiale. Utilizatorul de regulă nu se mulțumește să răspundă la câteva întrebări asupra naturii faptelor, el dorește în general să urmărească și raționamentul motorului. Din această cauză, există un modul special, conceput pentru a răspunde la motivul adresării unei întrebări sau la modul în care a ajuns sistemul la o anumită concluzie: *trasorul*. Pentru a utiliza cu încredere un sistem expert, trebuie să fim siguri că acesta este adecvat și de bună calitate. Din această cauză, sistemele, înainte de a trece în faza de exploatare, trec printr-o fază de

dezvoltare și implementare. Modulul folosit pentru dezvoltare, permițând îmbogățirea și validarea sistemului, este de obicei tot traserul. Acesta poate fi ajutat și de către proceduri sau module care nu servesc în cazul exploatării normale (pentru a desemna ansamblul acestor instrumente se utilizează expresia: *motor de elaborare*).

Noțiunea de motor de elaborare acoperă proceduri asociate traserului care se folosesc de regulă numai în modul de dezvoltare. Motorul coordonează activitatea traserului și a editorului, permițând facilitarea dezvoltării. Această dezvoltare înseamnă printre altele îmbogățirea bazei de reguli (prin adăugări), validarea regulilor (verificarea corectitudinii raționamentelor). Într-un motor cu coeficienți de verosimilitate trebuie realizată și ajustarea coeficienților. Calitatea unei baze de reguli poate fi măsurată prin considerarea următoarelor criterii: completitudinea, coerența și, în sistemele cu coeficienți de certitudine, neredundanța. Completitudinea înseamnă că sistemul nu riscă să se oprească datorită inexistenței unor reguli în baza de cunoștințe. Ea este foarte importantă în sistemele industriale cu buclă închisă, altfel fiind necesară intervenția utilizatorului. Coerența reprezintă proprietatea unei baze de reguli de a produce fapte necontradictorii, pentru orice bază de fapte necontradictorii.

Există două noțiuni de învățare care sunt asociate sistemelor expert: achiziționarea noilor reguli și reperarea celor euristice performante. Pentru reguli, problema se pune în funcție de scopul urmărit. Putem încerca să adăugăm reguli într-o structură de cunoștințe deja stabilită sau putem să dorim achiziționarea regulilor prin niște exemple (inducție). În primul caz ar fi vorba mai ales de îmbogățire, fapt ce ține de motorul de inferență. În al doilea caz, pentru exemplele bine structurate, se regăsește noțiunea de tabelă de decizie (o tehnică răspândită deja în anii '60, redevenită modernă odată cu sistemele expert). O asemenea tabelă poate fi interpretată și ca o matrice de exemple sau de situații. În sistemele obișnuite (cu inducție) se folosesc asemenea tabele pentru generarea ansamblurilor de reguli care să conducă la aceleași concluzii ca și matricea, pentru situațiile pe care aceasta le conține.

Există și alte tehnici, mai complicate, dar care se bazează pe structurarea avansată a tipurilor de reguli. Anumite sisteme sunt capabile să înlocuiască într-o regulă o constantă cu o variabilă sau să abandoneze o parte a premisei pentru obținerea unor reguli mai puternice (generalizare). Învățarea poate să se refere și la structura de control. Aceasta înseamnă posibilitatea asocierii unor legi euristice performante sau a unor metareguli stărilor din baza de fapte, după ce sistemul a constatat că conjuncția acestor stări și a acestor euristici ar putea conduce în mod rapid la rezultat.

Considerăm că în acest domeniu există probabil perspectivele cele mai promițătoare. Totuși, sistemele de învățare sunt considerate în general încă obiecte de laborator, achiziționarea cunoștințelor făcându-se de obicei fără învățare propriu-zisă.

4.3.1.15. Utilizatorii unui sistem expert

În general un sistem expert are două mari clase de utilizatori: utilizatorii obișnuiți și personalul specializat. Utilizatorii obișnuiți așteaptă de la sistem să fie mult mai prietenos decât personalul specializat; acesta din urmă tinde să fie orientat spre detalii, dorește să folosească un sistem complex în munca lor zilnică și sunt interesați în creșterea eficienței sistemului. În multe cazuri analiștii sunt cei care intermediază între utilizatorii obișnuiți și sistemul expert. Rolul intermediarului în folosirea unui sistem expert este ceva obișnuit și merită o atenție separată. Utilizarea unui intermediar permite utilizatorului să beneficieze de facilitățile oferite de sistem fără însă a intra în contact cu tastatura. În practică se întâlnesc trei categorii de intermediari ce reflectă tipuri diferite de sprijinire a utilizatorului:

-*asistentul* care este o persoană specializată în probleme manageriale și are experiență cu privire la asistarea deciziilor.

-*utilizatorul expert* ce deține cunoștințe în aplicarea unuia sau mai multor tipuri de instrumente specializate în "problem solving".

-*analistul de sistem* are cunoștințe generale despre domeniul aplicației, despre administrarea sistemului și cu abilități considerabile în construirea sistemelor.

Personalul specializat (utilizatorii experți) caută un răspuns la problema cercetată dar pot realiza și perfecționarea cunoștințelor lor.

4.3.1.16. Sisteme expert bazate pe fapte și reguli incerte

Cunoștințele experților sunt afectate de ambiguitatea și imprecizia limbajelor naturale. În plus, subiectivismul diferiților experți este un factor important, de care trebuie să se țină cont. Prin urmare baza de cunoștințe a unui sistem expert nu este în mod necesar nici total completă nici total consistentă. Incertitudinea informațiilor conținute în baza de cunoștințe necesită ca motorul de inferență să fie capabil să tragă concluzii din acest fel de informații.

Marea majoritate a analizelor și deciziilor implică, inevitabil, un anumit grad de incertitudine. Însăși specificul domeniului de activitate reflectat își pune amprenta, existând domenii care operează cu un grad mai mare de incertitudine decât altele. Spre exemplu, faptele și regulile necesare pentru definirea configurației unei instalații industriale se deosebesc, considerabil, din acest punct de vedere, de cele care pot fi formulate pentru stabilirea unui diagnostic medical sau a gestiunii unui portofoliu.

Prima modalitate de considerare a incertitudinilor și/sau aproximațiilor în cadrul sistemelor expert este introducerea acestora în obiectele cărora li s-a atribuit o valoare. Acestea pot avea atât valori numerice (sub formă de fracții, procente etc) cât și valori lingvistice (acestea fiind evaluate de regulă indirect). Se pot introduce probabilități și statistici fără probleme atât în fapte cât și în reguli. În fond, este de ajuns să discretizăm scările de probabilități și de valori (adică să considerăm un număr finit de trepte) pentru a putea lua în calcul toate situațiile posibile, scriind un număr finit de reguli. Acestea presupune introducerea unor cunoștințe de genul:

Dacă $0,70 \leq \text{PROB} \leq 0,80$
Atunci $\text{PROB} = 0,75$.

Abordarea se poate simplifica prin utilizarea variabilelor. Aceste considerații simpliste (dar operaționale) nu sunt întotdeauna satisfăcătoare, în special în domeniul diagnosticării. Acesta este motivul pentru care multe sisteme expert (ex. MYCIN, PROSPECTOR) au introdus noțiunea de *inferență incertă*. Să considerăm următoarea inferență:

Dacă F_1 și F_2 și F_3 atunci E_k
unde F_1, F_2, F_3 sunt fapte (observate sau demonstrate) și E_k este un diagnostic (sau un fapt dovedit). Atunci când evenimentul $F = \{ F_1, F_2, F_3 \}$ este observat, probabilitatea ca E_k să fie concluzia cea potrivită dintre E_1, E_2, \dots, E_n concluzii disjuncte posibile, este dată de formula lui Bayes^{84 85}:

$$P(E|F) = \frac{P(F|E_k) \cdot P(E_k)}{\sum P(F|E_i) \cdot P(E_i)}$$

În această expresie, $P(F|E_k)$ este probabilitatea de observare a lui F atunci când este formulat diagnosticul E_k (când E_k este adevărat). Aceasta este o probabilitate "a priori" care poate fi dedusă din serii statistice sau dintr-o estimare precedentă a expertului uman. Atunci când probabilitățile se bazează pe estimarea unui expert uman, este vorba de *probabilități subiective*. Formula lui Bayes poate fi interpretată și ca transformarea probabilității apriorice $P(E_k)$ în probabilitatea posterioară $P(E_k|F)$ prin observarea lui F . Acest model prezintă însă mai multe deficiențe criticabile, astfel⁸⁶:

-precizarea probabilităților apriorice $P(E_k)$ este dificilă, cu atât mai mult cu cât subiecții de cele mai multe ori nu raționează în domeniul incertitudinii cu probabilități subiective (după cum dovedesc experimentele)

⁸⁴ [Carlin00] Bradley P. Carlin, Thomas A. Louis –Bayes and Empirical Bayes Methods for Data Analysis –Second Edition Chapman & Hall, CRC, Boca Raton, 2000.

⁸⁵ [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.

⁸⁶ [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.

-nu se poate calcula $P(F|E_k)$ din $P(F_i|E_k)$ dacă nu se acceptă ipoteza independenței condiționale a faptelor F_i . Această ipoteză este aproape întotdeauna admisă în practică, deși are puține șanse de a fi satisfăcută în realitate.

Cele mai multe din sistemele expert actuale folosesc teoria bayesiană pentru modelarea incertitudinii. Astfel concluziilor sistemului le sunt asociate factori de certitudine (sau de incertitudine) probabilistică. Factorul de certitudine este o valoare numerică atașată unei premise sau unei concluzii dintr-o regulă și reprezintă gradul de încredere în acel enunț. Factorii de incertitudine sunt adesea atribuiți de către experți în baza raționamentelor lor specifice. În realitate nu orice tip de incertitudine poate fi tratat în manieră probabilistică. Incertitudinea asociată opiniilor diferite ale membrilor grupului de experți nu poate fi modelată folosind factorul de incertitudine calculat cu formula lui Bayes.

O altă soluție constă în introducerea incertitudinii sau aproximației chiar în exprimarea cunoștințelor (de exemplu se folosesc formulări de tipul “Este probabilă creșterea prețului acțiunilor X” sau “Nivelul inflației se va situa în jurul valorii de Y%”). Această soluție este insuficientă în multe cazuri, ceea ce a dus la căutarea altor modalități de reprezentare și tratare a incertitudinii. Una din tehnicile folosite constă în a atribui elementelor din baza de cunoștințe un coeficient destinat să exprime gradul de siguranță al acestora, denumit coeficient de certitudine (CC)⁸⁷. Acest coeficient poate lua valori cuprinse între 0 și 100, 0 corespunzând valorii fals și 100 valorii adevărat din logica binară. Dacă CC -ul unui fapt este de 70 înseamnă că faptul respectiv este cert în proporție de 70%. Coeficienții de certitudine nu sunt probabilități nici din punct de vedere conceptual și nici matematic și, în consecință, gradul de certitudine al tuturor elementelor dintr-un context dat nu trebuie să dea o sumă egală cu 100.

Tratarea incertitudinii presupune, în aceste condiții, atât evaluarea gradului de incertitudine al faptelor de bază cât și combinarea valorilor CC pentru evaluarea gradului de certitudine al faptelor compuse și a celor deduse prin aplicarea regulilor.

Coeficienții de certitudine ai faptelor compuse se determină, în funcție de tipul conectorilor logici, pe baza relațiilor următoare:

$$CC(A \text{ and } B) = \min(CC(A), CC(B))$$

$$CC(A \text{ or } B) = \max(CC(A), CC(B))$$

$$CC(\text{not } A) = 100 - CC(A)$$

unde $CC(A)$ și $CC(B)$ desemnează coeficienții de certitudine ai faptelor A și B.

Regulile pot avea la rândul lor coeficienți de certitudine, care exprimă cât de certă este concluzia dedusă pe baza premiselor lor. De exemplu⁸⁸:

R1. CC=70

Dacă X aparține ramurii Y

și Ramura Y tinde să crească

Atunci Cumpără acțiuni emise de X.

Gradul de certitudine al concluziei “Cumpără acțiuni emise de Y” dedus din cele două premise este 70.

Aceeași concluzie poate fi dedusă dintr-o regulă cu un coeficient de certitudine diferit:

R2. CC=85

Dacă Cifra de afaceri a societății X este stabilă

și Valoarea acțiunilor emise de X tinde să crească

Atunci Cumpără acțiuni emise de X.

Având în vedere că și premisele pot avea la rândul lor CC, în cazul general:

CC=c

Dacă premise

Atunci Concluzie.

⁸⁷ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

⁸⁸ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

coeficientul de certitudine al concluziei se stabilește cu relația:

$$CC(concluzie) = CC(premise) \times c / 100$$

În situațiile în care aceeași concluzie poate fi dedusă din aplicarea mai multor reguli diferite, QC al unei asemenea concluzii se obține prin combinarea coeficienților returnați de fiecare regulă care o cuprinde:

$$CC(R1, R2) = CC(R1) + CC(R2) - (CC(R1) \times CC(R2) / 100)$$

unde $CC(R1)$ și $CC(R2)$ sunt coeficienții de certitudine ai aceleiași concluzii deduse prin regulile $R1$ și $R2$.

Pentru exemplificarea aspectelor teoretice anterior prezentate considerăm următorul exemplu⁸⁹.

Presupunem o bază de fapte ce conține următoarele informații:

Ramura telecomunicațiilor tinde să crească	(CC=75).
Ramura siderurgică nu tinde să crească	(CC=80).
ALFA aparține ramurii telecomunicații	(CC=100).
BETA aparține ramurii siderurgice	(CC=100).
Cifra de afaceri a societății ALFA este stabilă	(CC=65).
Cifra de afaceri a societății BETA este stabilă	(CC=70).
Valoarea acțiunilor emise de ALFA tinde să crească	(CC=90).
Valoarea acțiunilor emise de BETA tinde să crească	(CC=60).

Aplicarea regulilor $R1$ și $R2$ conduce la următoarele rezultate pentru coeficienții de certitudine:

-pentru societatea ALFA

$$CC(R1) = CC(premise) \times 70/100 = \min(75, 100) \times 70/100 = 52,5$$

$$CC(R2) = CC(premise) \times 85/100 = \min(65, 90) \times 85/100 = 55,25$$

$$CC(R1, R2) = 52,5 + 55,25 - (52,5 \times 55,25 / 100) = 78,75$$

deci $CC(\text{"Cumpără acțiuni emise de ALFA"}) = 78,75$.

-pentru societatea BETA

$$CC(R1) = CC(premise) \times 70/100 = \min(100-80, 100) \times 70/100 = 14$$

$$CC(R2) = CC(premise) \times 85/100 = \min(70, 60) \times 85/100 = 51$$

$$CC(R1, R2) = 14 + 51 - (14 \times 51 / 100) = 57,86$$

deci $CC(\text{"Cumpără acțiuni emise de BETA"}) = 57,86$.

Rezultă că investiția în acțiuni ALFA este mai oportună decât cea în acțiuni BETA.

Anumite sisteme au încercat să evite utilizarea formulei lui Bayes sau să o adapteze. În sistemul MYCIN s-a evitat utilizarea formulei lui Bayes și s-a folosit atât un coeficient de certitudine aplicat asupra regulilor cât și asupra faptelor.

Coeficienții de certitudine atribuiți regulilor și faptelor influențează considerabil rezultatele. Din această cauză este deosebit de important ca ei să fie corect evaluați în practică, ceea ce nu este întotdeauna simplu sau evident.

O altă abordare se bazează pe noțiunea de "posibilități". Aceasta se poate considera o generalizare a noțiunii de probabilitate⁹⁰. Obiectivul acestor modele este să schimbe logica "totul sau nimic" conținută în *modus ponens*.

În fine, rămân cunoștințele aproximative exprimate în valori lingvistice, de genul "Inflația este mare". Dacă este peste 15% ea este considerată mare, sub 10% nu este mare. Între aceste valori se poate

⁸⁹ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

⁹⁰ [Dubois88] Dubois D, Prade H –Possibility Theory –Approach to Computerized Processing of Uncertainty, Plenum Press, New York, 1988.

atribui un grad de apartenență faptului că inflația este mare, conform teoriei mulțimilor vagi a lui Zadeh. Manipularea unor astfel de funcții de apartenență în sistemele expert permite evitarea efectelor de prag (câteodată neplăcute în dirijarea procedurii). În practica dirijării (controlării) procedurilor, pragurile –mai mult sau mai puțin arbitrare- există, pentru că întotdeauna există un moment când operatorul decide cum să acționeze (fără să fi făcut ceva până atunci), această acțiune fiind de multe ori discretă. Preluarea cunoștințelor vagi și a incertitudinilor sub forme declarative nu este încă foarte răspândită din cauza modelelor care utilizează coeficienți de verosimilitate și de certitudine (dar își face simțită prezența din ce în ce mai mult).

În afara nivelului specific de certitudine sau incertitudine cu care manipulează fiecare domeniu de activitate practică, incertitudinea mai poate proveni într-un sistem expert și din absența unor informații (informație incompletă). Se consideră că remedierea lui solicită un efort adecvat de completare a bazei de cunoștințe. În cazul în care acest lucru nu este posibil, atunci trebuie conviețuit cu această incertitudine și astfel devine foarte importantă dezvoltarea unui sistem expert performant în soluționarea incertitudinii.

4.3.2. Elaborarea unui sistem expert

4.3.2.1. Considerații generale despre elaborarea unui sistem expert

În elaborarea unui sistem expert, punctul cheie îl reprezintă introducerea cunoștințelor în sistem. Pentru atingerea obiectivului pot fi folosite mai multe procedee, în funcție de complexitatea sistemului expert. Dacă sistemul expert este de tipul celor bazat pe reguli, atunci cunoașterea poate fi introdusă sub forma unei mulțimi de reguli. Dacă sistemul este de tipul celor producătoare de reguli, atunci regulile pot fi deduse din datele avute la dispoziție. Colectarea, modelarea respectiv introducerea cunoașterii nu este o operație atât de simplă pe cât ar părea la prima vedere.

Sistemul trebuie să dispună în același timp de seturi de reguli înlănțuite, cuprinzătoare, pentru acoperirea tuturor posibilităților și de o bază de cunoștințe bine structurată. Cunoștințele (și regulile) conținute sunt ierarhizate. La sistemele expert generate cu ajutorul mediilor de dezvoltare existente, cunoștințele, odată introduse și cuprinse în baza de cunoștințe, rămân nemodificate pe parcursul funcționării sistemului adică utilizatorul poate avea acces la cunoștințe prin chestionarea regulilor dar regulile nu se modifică. Există situații în care regulile sunt actualizate, completate sau îmbogățite dar aceste schimbări se realizează separat de interacțiunile obișnuite cu baza de cunoștințe.

Modulul de inferență –după cum sugerează și denumirea- este componenta răspunzătoare pentru modul de funcționare al sistemului. Mașina are acces la reguli, promovând astfel relațiile, legăturile, conexiunile și deducțiile necesare. Există două moduri principale, considerate ca fiind de bază pentru raționamente: deducția (permite să se obțină concluzii pornind de la premisele regulilor verificate) și inducția (conduce la regulile de verificat, pornind de la combinații ale factorilor, chiar la scheme predefinite ale acestora). Acestea pot fi aplicate analitic (se descompun raționamentele în submodule mai ușor de înțeles) și/sau sintetic (se reunesc elementele separate) asupra domeniilor considerate. De fapt, pentru rezolvarea sistemelor complexe, metoda clasică este prin segmentarea acestora. Un alt mod de raționare, apărut ceva mai târziu și destinat în principal eficientizării inducției, este abducția. Acest procedeu constă în realizarea și combinarea unor scheme de observare ipotetice pentru pornirea inducției. Abducția nu este o tehnică inferențială validă dar uneori este foarte necesară, mai ales în sistemele expert de tip diagnostic unde poate genera explicații posibile.

Natura și numărul regulilor nu influențează modul de funcționare al mașinii de inferență. Deci, dacă se impune, se pot modifica regulile fără ca aceasta să afecteze mașina de inferență deși raționamentele (drumul care conduce prin baza de cunoștințe) pot suferi schimbări.

Există mai multe moduri de raționament. Astfel, *iterația* constă în repetarea unei secvențe de raționamente până când condiția de oprire este îndeplinită, iar *recursivitatea* constă în apelarea unui raționament de el însuși, până când acest raționament se bazează pe o problemă simplă, pe care a soluționat-o. Astfel, funcțiunea de bază a sistemelor expert este *raționamentul*, deci putem defini un sistem expert ca “un program informatic capabil să reproducă raționamente umane”⁹¹.

⁹¹ [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.

Considerăm că este foarte important ca toate sistemele să poată fi utilizate foarte ușor. Erorile, de orice natură ar fi, trebuie să fie sesizabile și în caz de nevoie să existe un sistem de help. De regulă sistemele expert dispun de programe ajutătoare integrate, acestea făcând parte din interfața utilizator.

Sistemul expert poate furniza explicații referitoare la raționamentul efectuat de către el. Pe parcursul chestionării sistemului, utilizatorul poate solicita explicații referitoare la motivele drumului deductiv urmărit de către sistem, după cum poate fi solicitat un maestru de șah să-și explice mutările. Valoarea fundamentală a sistemului expert rezidă în capacitatea de a înlocui expertul uman. Ca urmare, considerăm că sistemele expert sunt realizate, în principiu, pentru a furniza recomandări, sfaturi și nu neapărat, răspunsuri și pot fi exprimate sumar prin ecuația:

$$\text{sistem expert} = \text{cunoaștere} + \text{inferență}$$

Principala caracteristică a sistemelor expert este derivată din baza de cunoștințe împreună cu un algoritm de căutare specific metodei de inferență. Deci un sistem expert poate fi definit ca un “program informatic capabil să reproducă raționamente umane”. El poate trata cu succes probleme pentru care nu există o soluție algoritmică clară. Programele tradiționale funcționează foarte bine în cazul problemelor repetitive. Sistemele expert funcționează altfel. Drumul care conduce prin baza de cunoștințe este determinat prin reacțiile utilizatorului la întrebările adresate de către sistem.

Denumirea de sistem expert se aplică de multe ori nu numai aplicației realizate ci și pachetului instrumental care ajută la elaborarea acestuia. Există o varietate relativ mare de medii accesibile pentru generarea unor sisteme expert (“*expert system shell*”, “*expert system toolkit*”) care permit dezvoltarea aplicațiilor chiar și pe microcalculatoare.

Tehnologiile de programare utilizate la dezvoltarea sistemelor expert pot fi privite și ca o evoluție a limbajelor calculatoarelor spre un nou nivel al exprimării (permițând utilizarea și reprezentarea cunoștințelor). Limbajele timpurii de programare sunt dificil de utilizat pentru nespecialiștii în calculatoare. Limbaje de programare mai recente (de nivel înalt, obiectuale, vizuale etc) sunt binevenite pentru utilizarea și includerea diverselor forme de exprimare umană. Sistemele expert fac parte deci dintr-o gamă de instrumente indispensabile pentru realizarea unor sisteme automate sau interactive, capabile să efectueze sarcini complexe. De aceea nu este cu puțință să se ignore toate raționamentele calitative sau simbolice care formează baza proceselor decizionale. Mai mult, aceste raționamente pot fi simulate prin sisteme expert. Încă de la început sistemele expert au avut asociate instrumente de cercetare operațională care au facilitat dezvoltarea unor sisteme de asistare foarte complexe (diagnosticare, interfețe inteligente, medii de producție etc).

4.3.2.2. Limbajele de programare folosite pentru elaborarea sistemelor expert

Evoluția informaticii a determinat și o evoluție în domeniul limbajelor de programare, existând în domeniul software o evoluție pe generații, paralelă cu cea a dezvoltării arhitecturilor hardware.

Prima generație a fost constituită din limbajele *cod-mașină*.

A doua generație a cuprins *limbajele de asamblare* care au reprezentat un pas înainte în sensul atribuirii de nume simbolice (mnemonici) secvențelor din cod binar.

A treia generație de limbaje a reprezentat un puternic salt calitativ, marcat de apariția *limbajelor de nivel înalt (limbajele procedurale)*. Dintre cele mai cunoscute amintim: FORTRAN, BASIC, COBOL, PASCAL și C. Aceste limbaje sunt în esență algoritmice (imperative), în sensul că programele scrise în unul din aceste limbaje cuprind comenzi ce descriu pas cu pas algoritmul de rezolvare a problemei.

Generația a patra de limbaje a adus cu sine răspândirea sistemelor pentru gestiunea bazelor de date, a limbajelor de interogare, generatoarelor de rapoarte și a limbajelor specializate pentru anumite domenii.

Se consideră că limbajele de *cincea generație* a *cincea* vor trebui să proceseze limbajul natural.

O categorie aparte o reprezintă *limbajele declarative (neprocedurale)* care descriu forma și natura rezultatelor împreună cu un set de informații și o metodă prin care rezultatul poate fi dedus din informațiile date. Acest stil nou de programare a fost denumit *programare logică*, întâlnită în literatura de specialitate și sub numele de *programare declarativă*. Debutul programării logice poate fi atribuit lui R. Kowalski și A. Colmerauer și se situează la începutul anilor '70.

În figura 29 ilustrăm limbajele ce pot fi folosite pentru elaborarea sistemelor expert, privite din perspectiva timpului și a gradului de ușurință în proiectare.

Există două limbaje care ar putea revendica rolul de limbaje de inteligență artificială: PROLOG și LISP. Acestea prezintă avantajul posibilității programării la nivel înalt, permițând concentrarea eforturilor asupra descrierii relațiilor ce apar în problema dată și nu irosirea eforturilor spre construirea unei “logici a calculatorului” (ceea ce se face într-un limbaj procedural). Obiectivul urmărit este simularea unor aspecte ale gândirii umane. Acest scop impune în majoritatea cazurilor strângerea și introducerea unei cantități cât mai mare de informație, ceea ce se realizează de obicei sub forma bazelor de cunoștințe.

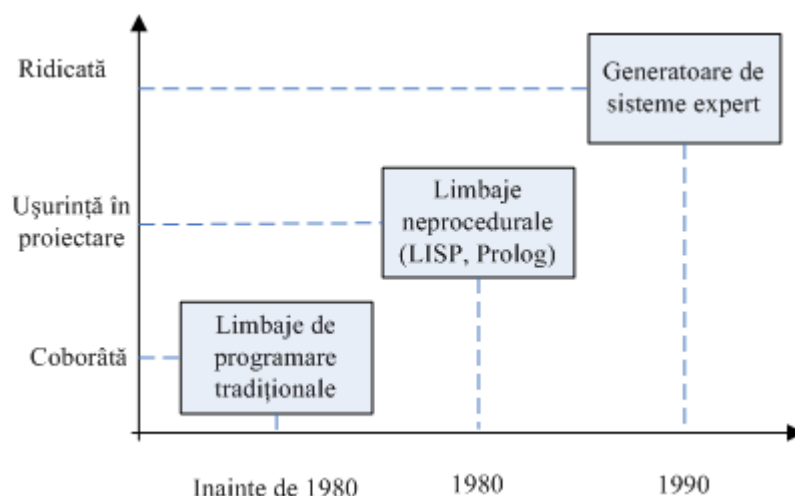


Fig. 29 Limbaje folosite pentru elaborarea sistemelor expert

Câteva din caracteristicile programării logice sunt următoarele:

Programarea logică este *descriptivă*. În locul unei serii de pași care să specifice cum să lucreze calculatorul pentru a rezolva problema, vom da doar descrierea problemei. Descrierea unei probleme presupune specificarea a trei componente:

- numele și structurile obiectelor implicate în problemă
- relațiile cunoscute care există între obiecte
- aspectele concrete (stările) descrise de aceste relații (care se pot specifica sub forma unor fapte și reguli)

Descrierea în programarea logică este folosită mai ales pentru specificarea relației dorite dintre datele introduse și rezultatele produse de aceste intrări.

Programarea logică folosește *fapte și reguli*. Limbajele de programare logică, în general, *au o sintaxă simplă și sunt concise*. Este mai ușor să se învețe astfel de limbaje decât limbajele de programare tradiționale care au o sintaxă mult mai stufoasă.

Este clar că sarcina programatorului este mult simplificată în cazul programării declarative, după specificarea scopului, sistemul preluând controlul și rezolvând problema cu ajutorul metodelor deductive de care dispune.

Limbajul care poate fi utilizat pentru a scrie un sistem expert nu depinde neapărat de forma adoptată pentru reprezentarea cunoștințelor. Au fost și sunt elaborate sisteme expert în BASIC, COBOL, FORTRAN, APL, PL/1, PASCAL, LISP, PROLOG, C etc. Primele cer din partea utilizatorului mare atenție în definirea cerințelor; acestea nu furnizează subsisteme necesare dezvoltării sistemelor expert.

Ele pot fi folosite cu rezultate satisfăcătoare dacă:

- aplicația nu evoluează rapid
- performanța este crucială
- nu există constrângeri de timp în implementarea aplicației

Limbajul *PROLOG* (PROgrammation en LOGique)⁹² este cel care a fost ales de către japonezi pentru programul de dezvoltare a calculatoarelor din generația a cincea. A fost dezvoltat mai întâi la Universitatea Luminy din Marsilia de A. Colmerauer și Ph. Roussel (în perioada 1970-1975), apoi la

⁹² [Clocksin81] Clocksin W.F., Mellish C.C. –Programming in Prolog, Springer-Verlag, New York, 1981.

Universitatea din Edinburgh. Este un limbaj bazat pe logica predicatelor. Acest limbaj este prototipul limbajelor declarative, prin opoziție cu programarea procedurală (limbajele procedurale sunt cele în care programatorul descrie pas cu pas un algoritm, precizând prin înșiruirea instrucțiunilor calea exactă a programului). Programarea declarativă constă în furnizarea “dezordonată” a datelor și regulilor în cadrul programului, iar programul “se descurcă” cu ajutorul mecanismului de inferență încorporat.

PROLOG are o caracteristică ce merită a fi relevată: regulile și faptele sunt amestecate și urmează exact același formalism, neexistând diferență între ele. Un fapt poate fi tratat ca o regulă fără precondiție.

Timpul rămâne însă dușmanul principal al programării declarative, deci atunci când o parte din prelucrarea unei probleme, chiar într-un sistem expert, conduce la un algoritm bine definit, nu trebuie să ezităm să includem această procedură în sistemul expert.

Există mai multe versiuni ale limbajului (variante cu restricții, extensii ale limbajului, combinații cu alte limbaje etc), precum și adevărate medii bazate pe acest limbaj, cu strategii de control diferite. Au fost și sunt tentative de a îmbina avantajele LISP-ului cu cele ale PROLOG-ului. Astfel au apărut LOGLISP (1980), QLOG (1982), FROG (1985) etc.

Limbajul *LISP* (LIST Processing) a apărut în anul 1959 la MIT (Massachusetts Institute for Technology SUA) sub coordonarea profesorului J. McCarthy. Primul manual LISP a apărut în 1962. Pentru marile universități americane, limbajul inteligenței artificiale și al sistemelor expert este LISP. Multe generatoare de sisteme expert funcționează cu interpretoare sau compilatoare LISP. Limbajul LISP este mai mult funcțional decât relațional sau logic, operațiile de logică (unificarea de exemplu) când sunt necesare, trebuie să fie programate, ceea ce constituie un impediment.

Autodesk a ales LISP ca limbaj de programare în AutoCAD pentru că:

- un interpretor LISP este ideal pentru prelucrarea datelor nestructurate care caracterizează proiectarea asistată de calculator;
- poate lucra cu colecții de obiecte eterogene, în grupuri de dimensiuni diferite;
- dintre toate limbajele de programare, LISP-ul este ușor de învățat și de utilizat;
- deoarece LISP are o sintaxă simplă, un interpretor LISP este ușor de implementat și necesită puțină memorie.

În acest limbaj se manipulează *atomi* și *liste*. Fiecare listă este organizată sub formă de arborescență binară –de exemplu, operația $1 \times (2 + 3)$ se va scrie:

$(\times 1 (+ 2 3))$. Primul element al listei este interpretat ca un operator (sau ca o specificație de funcție) care se aplică elementelor următoare din listă. Scrierea regulilor este ușoară, ca de altfel și prelucrarea și compararea listelor, ceea ce este foarte util pentru manipularea arborescențelor și pentru operația de filtrare. LISP nu face deosebire dintre proceduri și date. Limbajul are o modularitate ridicată, adăugarea de reguli și de cunoștințe se face foarte ușor ceea ce reprezintă un atu pentru realizarea sistemelor expert. Performanțele mediocre ale interpretoarelor și compilatoarelor LISP pe calculatoare obișnuite, au determinat realizarea unor calculatoare dedicate (din punctul de vedere al timpului de răspuns). Aceste calculatoare dedicate (conținând “procesoare LISP”) au cedat însă progresiv locul stațiilor de lucru care vizează publicul larg și, care datorită progresului microprocesoarelor, au performanțe comparabile cu cele ale calculatoarelor LISP. Limbajul LISP are un mare dezavantaj ce constă în lipsa motorului de inferență care trebuie scris de programator, de fiecare dată când se construiește un nou sistem expert.

Perioada care a urmat este caracterizată de apariția unor limbaje ca: MICRO-PLANNER (1971), PLANNER (1972) și CONNIVER (1972).

Cu toate avantajele programării logice, limbajul care domină în prezent realizarea sistemelor expert este C++, un limbaj orientat pe obiecte. Cauza o reprezintă viteza de execuție (mult superioară limbajelor ce folosesc programarea declarativă), portabilitatea și răspândirea sistemului de operare UNIX. Cercetările continuă și azi pentru dezvoltarea limbajelor orientate pe obiecte cât mai interactive. Cel mai vechi limbaj obiectual este SmallTalk⁹³, realizat între anii 1970 și 1976 de către Kay, pornind de la limbajul SIMULA. Ulterior au apărut versiunile SmallTalk-72, SmallTalk-76, SmallTalk-80.

⁹³ [Goldberg83] Goldberg A., Robson D. –Smalltalk-80 –The Language and its Implementation, Addison-Wesley, 1983.

Limbajele orientate pe obiecte sunt foarte modulare (grație autonomiei obiectelor), ele se apropie de cadre și posedă proprietățile de moștenire și ierarhizare a claselor. Modularitatea este asigurată și de ușurința introducerii și distrugerii claselor și a obiectelor. Aceste limbaje sunt ușor lizibile pentru că toate aspectele referitoare la o clasă sau un obiect sunt regrupate izolat. Combinația reprezentării prin cadre și a limbajelor orientate pe obiecte a dat naștere unor limbaje destinate în special dezvoltării sistemelor expert (LOOPS, MERING, KOOL, LRO etc.) Anumite medii pentru dezvoltarea sistemelor expert sunt construite în jurul acestor limbaje (de exemplu BULL propune limbajul KOOL). Există și limbaje născute din extinderea limbajului LISP sau din combinarea acestuia cu limbaje orientate pe obiecte (FLAVORS, OBJVLISP, CEYX, ROSACE etc). Au apărut chiar medii orientate pe obiecte realizate în C (precum OBJECTIVE-C).

Reprezentarea cunoștințelor într-un sistem expert trebuie să satisfacă în principiu două exigențe contradictorii. Prima exigență este legată de sistemul de calcul. Programele sunt cu atât mai eficace cu cât limbajul în care sunt realizate este mai apropiat de limbajul mașină (prima versiune a lui NEXPERT pe Macintosh era realizată în Assambler, acum este în C). A doua exigență este transparența (lizibilitatea) bazei de cunoștințe. O bază de cunoștințe trebuie să poată fi citită, completată, modificată, întreținută de către un expert în domeniul desemnat (în general neinformatician). Limbajul cel mai bun pentru a răspunde acestei cerințe ar fi probabil limba maternă a expertului, la care s-ar adăuga jargonul de specialitate. Problema este că, din păcate, calculatorul nu înțelege limbajul expertului și invers. Reprezentarea cunoștințelor constă în găsirea unei terminologii intermediare, ceea ce nu poate fi decât un compromis (aceasta explică faptul că, pe de o parte sistemele expert sunt numeroase, pe de alta ele nu sunt niciodată unanim aprobate). Conform celor două exigențe majore amintite, există două tendințe.

Prima tendință este folosirea limbajelor de programare. Astfel, la scrierea unui program în PASCAL, se poate spune că programul va funcționa pe orice calculator și cu orice compilator PASCAL adaptat calculatorului (cu excepția câtorva aspecte legate în majoritate de intrări/ieșiri). În ceea ce privește limbajele, ordinea lor de preferință ar fi: LISP, PROLOG și limbaje orientate pe obiecte. Acestea însă sunt adaptate mai degrabă informaticienilor și calculatoarelor decât expertului uman.

Cealaltă tendință este apelarea la instrumente de dezvoltare și generatoare de sisteme expert (instrumente special concepute pentru a scrie baze de cunoștințe). După cum indică și denumirea "instrument", acestea nu au o destinație universală, spre deosebire de limbajele care permit exprimarea mai multor lucruri. Aceste pachete de programe (care într-o anumită măsură sunt ca niște limbaje limitate) sunt puternic dependente de interpretor. Capacitatea de adaptare la necesitățile utilizatorului, în cazul generatoarelor este foarte mare.

Cele două tendințe (ilustrate în figura 30) tind să se reunească, în măsura în care limbajele oferă medii de dezvoltare, adică mai multe instrumente asociate, care furnizează aproape același funcționalități ca și generatoarele de sisteme expert.

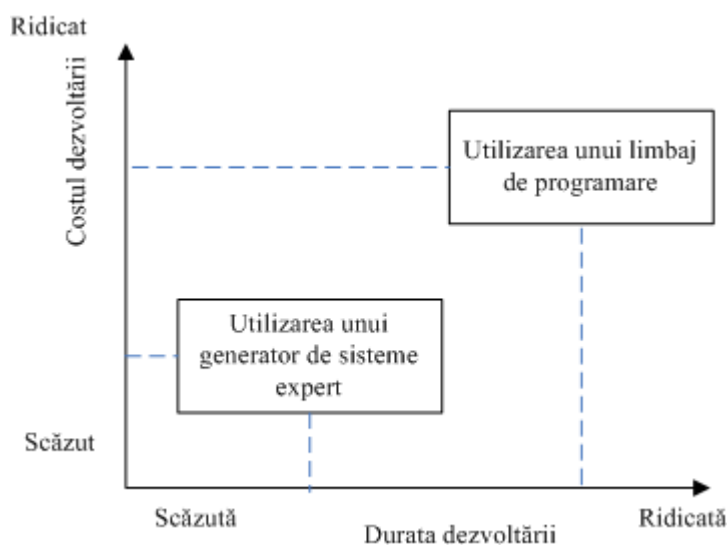


Fig.30 Soluții de dezvoltare a sistemelor expert

4.3.2.3. Metodologii de elaborare a sistemelor expert

Dicționarul Webster definește o metodologie ca:

- o serie de metode și tehnici intercorelate
- studiul unor metode.

Fiecare metodologie presupune un anumit mod de organizare a proceselor, procedurilor și șabloanelor necesare ducerii la îndeplinire a unui proiect. Scopul unei metodologii este prezentat grafic în figura 31.

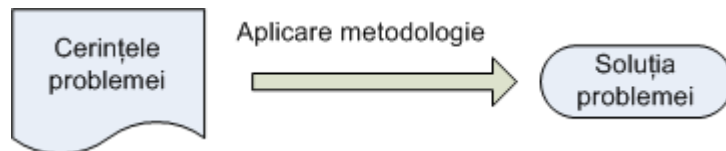


Fig. 31 Scopul unei metodologii

Figura 94 ilustrează conceptele cu care operează o metodologie⁹⁴:

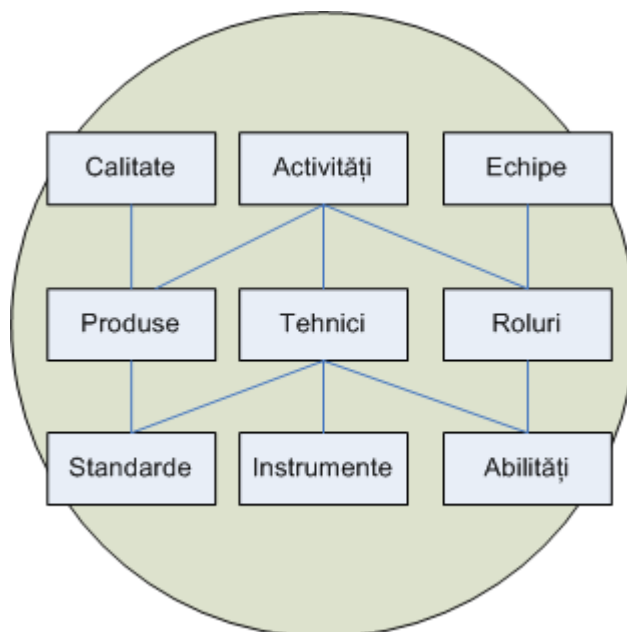


Fig. 32 Conceptele cu care operează o metodologie

O metodologie cuprinde mai multe componente⁹⁵:

- o definiție a spațiului problemelor pentru care acea metodologie este aplicabilă
- un set de modele care reprezintă diferite aspecte ale domeniului problemei sau ale soluțiilor la diferite nivele
- un set de metode care transformă instanțele unui model într-un alt model
- un ghid care definește o ordine în care se aplică pașii în vederea găsirii soluției la problema dată.

O metodologie stabilește anumite roluri și activități care se desfășoară pe parcursul derulării unui proiect. În cazul în care o persoană singulară derulează un proiect în conformitate cu o metodologie, ea va juca toate rolurile prevăzute în metodologie și va desfășura activitățile în conformitate cu ea.

Metodologiile de dezvoltare a sistemelor inteligente au înregistrat trei perioade evolutive:

1) *Perioada programării* este delimitată de sfârșitul anilor 1970 până la mijlocul anilor 1980, când s-au dezvoltat noile tehnici de programare. Sistemele au fost descrise folosind tehnici de reprezentare a cunoașterii utilizând reguli, cadre, rețele semantice etc. Mediile de dezvoltare apărute asigurau suportul utilizării acestor tehnici de reprezentare a cunoașterii. În acest sens sunt reprezentative mediile Art, Acquire, Exsys Professional, Guru, Level 5, Ops, VP-Expert etc., dezvoltate mai ales de firme americane și japoneze.

2) *Perioada modelării* începe în anul 1985 când Clancey a publicat lucrarea sa despre diagnosticul euristic (heuristic diagnosis) unde analizează un număr de sisteme din punctul de vedere al puterii de abstractizare și identifică metoda specifică de rezolvare a problemei care stă la baza comportamentului acestora. Metoda clasificării euristice, descoperită de Clancey este descrisă în termenii pașilor inferențiali și tipurilor de cunoaștere utilizate. Descoperirile lui Clancey au deschis calea apariției, la sfârșitul anilor 1980 a unor metodologii orientate spre cunoaștere.

⁹⁴ [Cockburn01] Alistair Cockburn –Agile Software Development, Addison Wesley, 2001 (adaptare)

⁹⁵ [Hußmann97] H. Hußmann: Formal Foundations for Software Engineering Methods. LNCS 1322, Springer-Verlag, Berlin, Germany, 1997

Modelarea servește obținerii caracteristicilor esențiale ale unui sistem, împărțindu-l în mai multe părți care vor fi mai ușor de înțeles și gestionat. Booch⁹⁶ afirmă că “un model este o simplificare a realității”. Sistemele existente în practică sunt niște entități uneori de dimensiuni apreciabile, având mai multe componente care acționează și conlucrează într-o manieră complexă. Modelarea ne ajută să înțelegem această complexitate, oferind diverse nivele de abstractizare și permițând în același timp comunicarea între mai multe persoane în cadrul unui anumit proiect. Fiecare persoană poate avea o altă viziune asupra sistemului iar modelarea oferă posibilitatea de a înțelege toate aceste viziuni într-o manieră unitară.

Procesul de modelare construiește modele conceptuale pentru diversele activități ce au la bază cunoașterea. În timpul fazei de achiziție a cunoștințelor, majoritatea acestora sunt nestructurate și oferite în modalități improprii gestiunii lor. Inginerul de cunoștințe, a cărui sarcină este crearea bazei de cunoștințe, încearcă să înțeleagă atât aspectul explicit cât și cel tacit al cunoștințelor și poate utiliza diverse diagrame vizuale pentru a stimula discuțiile cu experții. În urma acestor discuții, inginerul de cunoștințe va construi modelul conceptual. Modelarea va conduce la înțelegerea conceptelor și mecanismelor sistemului ce urmează a fi construit, cum ar fi sarcini, metode, inferențe etc.

3) *Perioada metodelor formale* își are debutul la începutul anilor 1980. Lipsa fundamentelor formalizării a condus spre cercetări susținute în scopul introducerii unor metode formale pentru a capta caracteristicile tuturor tipurilor de cunoaștere în procesul inferențial. Astfel, în Europa s-au înregistrat cele mai importante succese în privința modelării conceptuale de tip KADS⁹⁷.

Apariția acestei metodologii a pornit de la existența diferențelor între date și cunoștințe. Aceste diferențe fac inadecvată aplicarea metodologiilor de analiză și proiectare a sistemelor structurate în cazul sistemelor inteligente. Cu toate acestea mulți nu utilizează această metodologie pentru proiectarea sistemelor inteligente, preferând prototipizarea rapidă.

Noțiunea de *metodologie de dezvoltare a sistemelor expert* a apărut la finalul anilor '80 și începutul anilor '90. Atâta vreme cât sistemele expert au fost dezvoltate în laboratoarele de cercetare ca și prototipuri, nu a existat o nevoie clară a unei abordări sistematice a dezvoltării acestora. Odată cu creșterea dimensiunii și a complexității acestora, a apărut nevoia îmbunătățirii și sistematizării dezvoltării acestor tipuri de sisteme. Hoog⁹⁸ clasifică aceste metodologii în 3 categorii:

1) *Metodologii diagramate (cu căsuțe și săgeți) (“Boxes and Arrows”)* sunt caracterizate prin așa numitul ciclu de viață care poate fi reprezentat printr-o succesiune de activități desfășurate într-o anumită ordine. Această abordare nu dezvoltă suficient modul de definire al activităților, metodele și tehnicile ce trebuie utilizate. Abordările prin această metodologie răspund la întrebarea *ce?* dar nu și la întrebarea *cum?* pentru activitățile evidențiate prin căsuțe. Totuși, această metodologie acoperă toate activitățile importante care apar în dezvoltarea sistemelor expert.

Conform acestei metodologii, ciclul de viață aferent dezvoltării unui sistem expert poate fi reprezentat astfel (figura 33):

⁹⁶ [Booch99] Booch, G., Rumbaugh, J. and Jacobson, I. -The Unified Modelling Language User Guide, Addison Wesley, Reading, Massachusetts, 1999.

⁹⁷ [Andone03] Andone I –Metodologiile și mediile de dezvoltare a sistemelor inteligente. Experiența europeană – Simpozion Internațional “Specializare, Dezvoltare și Integrare”, Cluj-Napoca, 2003.

⁹⁸ [Hoog98] Hoog R.D. –Methodologies for Building Knowledge Based Systems: Achievements and Prospects, J. Leibowitz, în The Handbook of Applied Expert Systems, CRC Press, 1998.

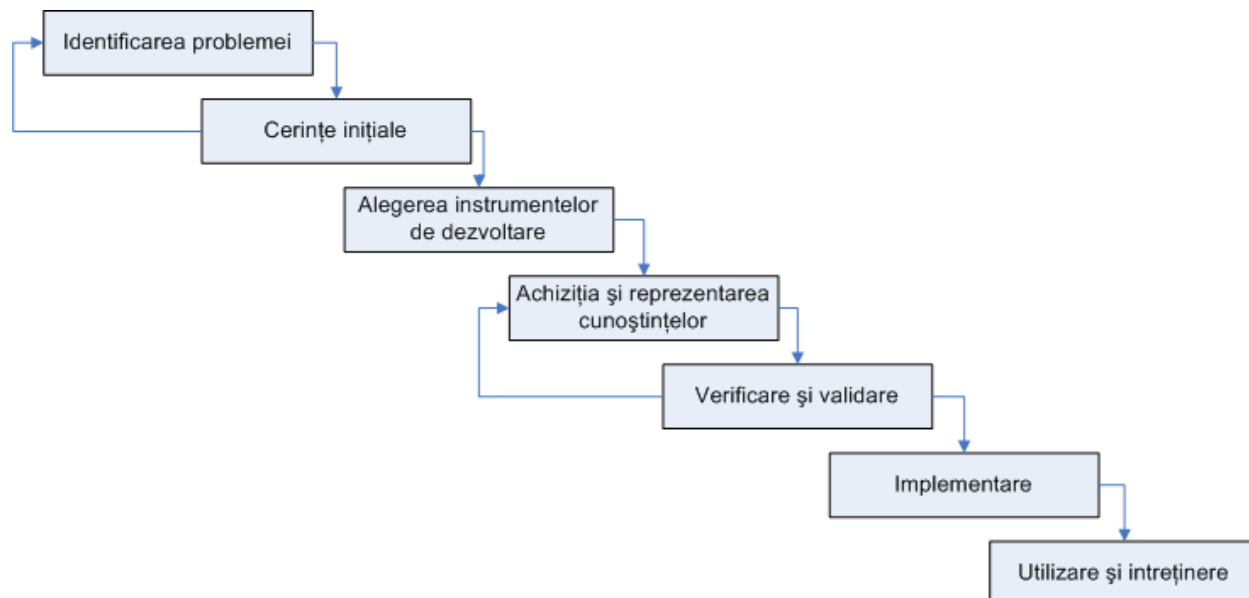


Fig. 33 Ciclul de viață al unui sistem expert folosind metodologii diagramate

Această metodologie prezintă mai multe abordări, diferențele constând în definirea diferită din punct de vedere semantic a etapelor, detalierea mai amănunțită a lor sau mai multe legături spre etapele anterioare, sporind prin acesta natura iterativă a acestei metodologii.

2) *Metodologia focalizată* se limitează la una sau mai multe metode sau tehnici părând mai degrabă o mini-metodologie. Această metodologie (ilustrată în figura 34) se axează în principal pe problema achiziției cunoștințelor care este abordată în 3 modalități:

- modelarea expertizei utilizând metode și tehnici de achiziția cunoștințelor din clasa celor manuale (precum interviurile, analiza protocolului, revizuirea cazurilor) și utilizând un inginer de cunoștințe. Această modalitate s-a folosit cu precădere în Europa
- transferul direct al cunoștințelor de la expert la baza de cunoștințe utilizând metode și tehnici adecvate de expertiză. Această modalitate încearcă să elimine cognitiicianul ca intermediar între expert și sistem (calculator).
- utilizarea unui mediu potrivit de achiziție de cunoștințe. Această cale are la bază ideea că achiziția cunoașterii operează cel mai bine când este suprapusă cât mai mult timp pe un domeniu particular al expertizei.

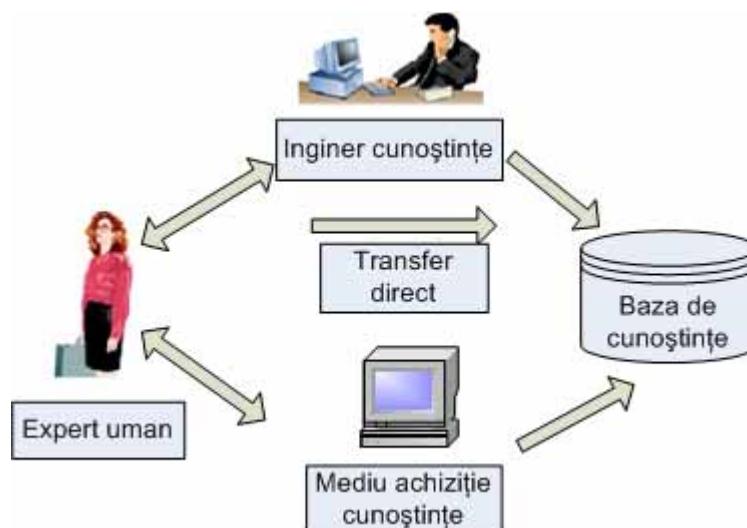


Fig 34 . Metodologia focalizată

3) *Metodologiile complet maturizate ("Full-Fledged")* cuprind două categorii:

a) metodologii pentru sisteme convenționale care au în componență și un sistem expert

b) metodologii destinate doar dezvoltării de sisteme expert

Principalul avantaj al acestor metodologii este acela că obligă dezvoltatorul unor astfel de sisteme să modeleze și să documenteze sistemul la nivelul cunoașterii înainte de a se ajunge la nivelul de implementare. Rezultatul va fi un model la nivelul cunoașterii, cunoscut și sub numele de modelul conceptual, permițându-se astfel analiza cunoașterii înainte de a decide un model particular de reprezentare a cunoștințelor în vederea implementării sistemului. Astfel se realizează independența activităților de analiză, specificare și proiectare de un anumit limbaj sau mediu de dezvoltare.

Metodologia KADS

Dintre metodologiile complet maturizate, cea mai completă și mai cunoscută este metodologia KADS, denumire a cărui acronim a evoluat de la “Knowledge Acquisition and Documentation System” și “Knowledge Acquisition and Design System” la “Knowledge Acquisition and Document Structuring”. KADS a fost dezvoltată de Wielinga și colaboratorii săi începând cu 1984 la Universitatea din Amsterdam și este rezultatul a două proiecte de cercetare ale Comunității Europene în cadrul unor programe Esprit, într-o cooperare cu alți parteneri din Europa. Versiunea actuală a acestei metodologii este denumită CommonKADS.

CommonKADS este o abordare structurată a dezvoltării sistemelor expert, venind în contrast cu abordările nestructurate cum ar fi prototipizarea rapidă. Ideea de bază a acestei metodologii este aceea de modelare. Analiza specifică ce va face sistemul, rezultând un model al comportării dorite a sistemului. Proiectarea va specifica arhitectura sistemului și cum componentele sistemului vor contribui la satisfacerea cerințelor. Metodologia cuprinde activitățile necesare a se desfășura în timpul analizei și proiectării sistemului. Metodologia CommonKADS are la bază următoarele principii:

- achiziția cunoștințelor necesită mai întâi modelarea lor
- părți ale expertizei pot fi captate și grupate în librării generice care pot fi reutilizate
- dezvoltarea unui sistem bazat pe cunoștințe trebuie să aibă la bază construcția unui model conceptual
- problema organizațională trebuie să fie parte a metodologiei
- procesul de dezvoltare trebuie să aibă la bază o abordare flexibilă, configurabilă, bazată pe analiza obiectivelor și a riscurilor.

Metodologia CommonKADS se bazează pe șase modele⁹⁹:

1. *Modelul organizației* se referă la problemele legate de mediul organizațional în care va funcționa sistemul. Acest model urmărește o organizație din cinci perspective majore: activități, structură, procese, autoritate și resurse.
2. *Modelul activităților* urmărește activitățile desfășurate pe parcursul diverselor procese. Dacă modelul organizațional indică faptul că o anumită sarcină poate fi automatizată, atunci se va genera un model al acesteia, urmărind detaliile activităților care rezolvă acea sarcină. Acest model determină și intrările și ieșirile fiecărui proces sau activități, capacitățile realizării unei anumite sarcini, frecvența sarcinii etc.
3. *Modelul agenților* reprezintă capacitățile cerute agenților pentru a îndeplini anumite sarcini și constrângerile performanței lor. Termenul de agent se referă la orice entitate care are de efectuat sarcini.
4. *Modelul expertizei* cuprinde cunoașterea utilizată de sistem pentru soluționarea sarcinilor. O facilitate distinctă a acestui model este utilizarea unor componente din modelul bibliotecilor. Metodologia CommonKADS oferă trei niveluri în vederea modelării expertizei¹⁰⁰. Primul nivel, denumit “*nivelul cunoașterii*” definește obiectele statice (concepte, proprietăți și relații) ale domeniului aplicației. Conceptele sunt clase care reprezintă obiecte, proprietățile sunt atribute ale conceptelor, iar relațiile leagă conceptele între ele. Al doilea nivel, “*nivelul de inferență*”

⁹⁹ [Schreiber94] Schreiber A. T., Wielinga B. J., ș.a. -CommonKADS: A comprehensive methodology for KBS development, IEEE Expert 9,6, 1994.

¹⁰⁰ [Kingston92] Kingston J. -Pragmatic KADS: a methodological approach to a small knowledge-based systems project, Expert Systems: The International Journal of Knowledge Engineering, 1992.

definește primii pași de inferență ce trebuie executați pe obiectele nivelului precedent pentru a deriva noi obiecte. Al treilea nivel, “*nivelul proceselor*” definește cum trebuie utilizată inferența pentru a rezolva problema.

5. *Modelul comunicației* realizează interacțiunea cu utilizatorul și cu alte sisteme software. De asemenea, el cuprinde comunicările dintre agenți din timpul proceselor, indicând toate tranzacțiile ce pot avea loc între agenți în timpul proceselor.
6. *Modelul bibliotecilor* oferă o legătură între modelul conceptual (sarcini, agenți, comunicare, expertiză) și sistemul ce urmează a fi implementat. Acest model promovează eficiența dezvoltării prin reutilizarea unor biblioteci deja create în timpul proiectării.

Figura 35 prezintă modelele metodologiei KADS și legăturile dintre acestea.

CommonKADS oferă atât limbaje formale cât și informale cu ajutorul cărora să se descrie aceste niveluri. În general, acestea se descriu informal la început, apoi se formalizează și rafinează în vederea implementării.

Dezvoltarea unui sistem expert necesită utilizarea unor metode formale. Unui sistem i se pot formaliza cerințele, conducând spre specificațiile sistemului care apoi vor fi implementate folosind limbaje de programare și demonstrate utilizând metode formale. Există cel puțin doi factori care explică interesul crescând pentru metodele formale. Primul este dat de evoluția tehnologiei: utilizarea diverselor variante de logici formale în limbajele de reprezentare a cunoștințelor (cum ar fi Prolog) cât și limbajele de specificare formală pentru aplicațiile tradiționale (cum ar fi Z sau VDM) sugerează o convergență a abordărilor. Al doilea factor este dat de faptul că mulți dintre avocații abordărilor formale ale dezvoltării sistemelor expert sunt preocupați de importanța acestor sisteme în domenii critice precum cel spațial sau nuclear.

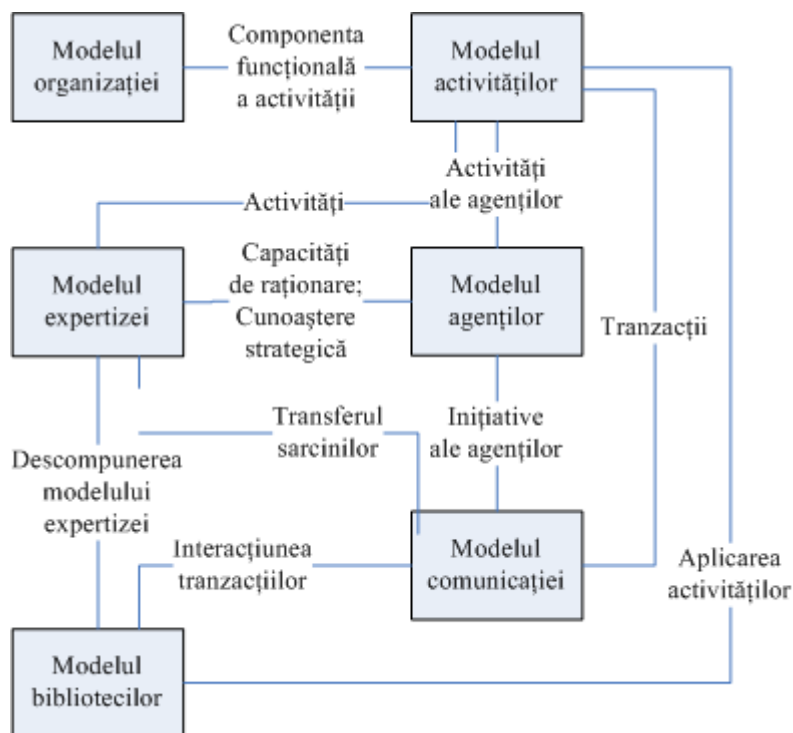


Fig. 35 Modelele metodologiei CommonKADS

Din acest punct de vedere, autorul consideră că există diferențe esențiale între sistemele expert și sistemele convenționale. Pentru aplicațiile tradiționale, metodele formale sunt utilizate pentru a crea un set inițial de specificații, care vor fi rafinate pe niveluri succesive în vederea implementării. Corectitudinea implementării este asigurată în mod similar unei demonstrații matematice: specificațiile inițiale sunt corecte și fiecare pas în procesul de rafinare urmează niște reguli bine definite. Acest lucru nu este întotdeauna adevărat pentru sistemele expert create îndeosebi pentru acele domenii care nu au fundamentată o teorie formală (de exemplu domeniul afacerilor sau medical). Pentru aceste domenii, o specificare formală corectă a cerințelor utilizatorilor este foarte greu de creat și o specificare completă

poate fi imposibilă. Cu toate acestea, din punctul de vedere al celor care dezvoltă astfel de sisteme¹⁰¹, o formalizare crescândă nu este de dorit. Dezvoltarea unui sistem expert în domeniul afacerilor, de exemplu, este atractivă tocmai datorită lipsei acestei formalizări, permițând în același timp modificarea cerințelor pe măsura evoluției acestora în timp. Înghețarea scopului unui proiect prea devreme poate conduce la eșecul lui. Folosind sistemele expert, putem investiga probleme și sarcini care nu sunt înțelese prea bine, cel puțin la început. Poate dacă le-am înțelege de la început, nu am mai utiliza un sistem expert ci am utiliza un algoritm ce poate fi transpus printr-o aplicație tradițională.

Dezavantajul metodologiei CommonKADS este dat de necesitatea întreținerii modelului conceptual, care poate îngreuna dezvoltarea sistemului, fiind consumatoare de timp. Cerințele se pot schimba rapid ceea ce antrenează uneori modificări în modelul conceptual. Recunoașterea acestui dezavantaj de către susținătorii acestei metodologii a condus spre dezvoltarea unor medii care să convertească rapid modelele date de această metodologie în prototipurile unor sisteme executabile.

Metodologia CommonKADS a devenit un standard în ceea ce privește modelarea cunoștințelor și este utilizat extensiv în proiectele de cercetare pe această temă.

Metodologia ORSA

Metodologia ORSA (Operational Research and Systems Analysis) pornește de la obiectivele generale ale proiectului, încercând să realizeze transformarea acestor obiective în cerințe pentru realizarea sistemului. Această metodologie consideră următoarele activități necesare realizării unui sistem inteligent (figura 36):

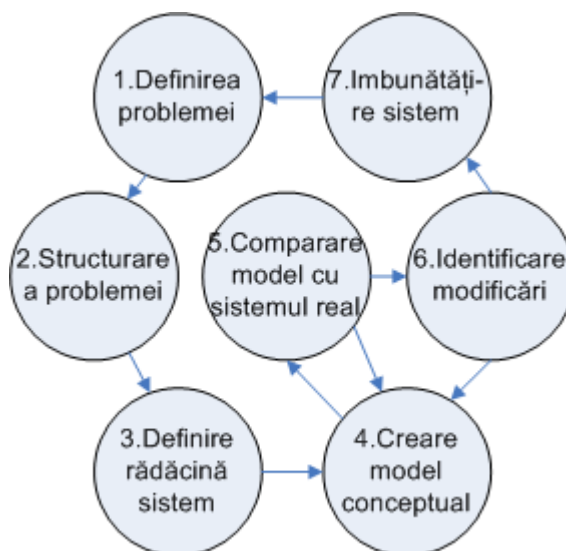


Fig.36 Schema de lucru generală în cadrul metodologiei ORSA

Definirea și structurarea problemei au ca scop formularea unei descrieri detaliate a problemei, rezultând o diagramă cu elemente de structură și procese, diagramă ce poate fi construită gradat și rafinată dacă în cadrul pasului 6 apar modificări posibile și dezirabile. Pasul 3 reprezentând elaborarea (fundamentarea) *rădăcinii* sistemului reprezintă o declarație a viziunii sistemului în legătură cu ceea ce trebuie el să realizeze. În acest sens, Checkland P.¹⁰², a definit o mnemonică denumită CATWOE cu următoarea semnificație:

C (Customer)	Ce este utilizatorul, pentru cine acționează sistemul?
A (Actors)	Cine îndeplinește principalele acțiuni din sistem?
T (Transformation)	Care este activitatea fundamentală desfășurată în sistem?
W (Wetanschaung)	Care este perspectiva care face această definiție să fie fundamentală?

¹⁰¹ [Keefe93] O'Keefe R.M., O'Leary D.E. -A review and survey of expert system verification and validation, Artificial Intelligence Review 7:1, 1993.

¹⁰² [***] Creative Tools and Techniques CATWOE <http://www.mycoted.com/creativity/techniques/catwoe.php>

O (Owners) Care sunt persoanele care decid asupra existenței sistemului?

E (Environment) Care sunt limitele sistemului?

În cadrul etapei a 4-a se obține un set de activități minim necesare pe care sistemul trebuie să le efectueze. Acestea sunt comparate cu cele din sistemul real, putând rezulta anumite modificări ale modelului conceptual. În ultima etapă, aceste modificări sunt puse în aplicare după care procesul se poate relua sau opri.

Metodologia RUDE

Metodologia RUDE (Run-Understand-Debug-Edit) a fost elaborată de D.Partridge, pe baza unei analize comparative a sistemelor inteligente cu cele convenționale. Sistemele convenționale (clasice) se pot caracteriza prin modelul “problemei complet formulate”. Conform lui Partridge, acest lucru nu se poate realiza în cazul sistemelor inteligente. Această metodologie cuprinde următoarele etape (figura 37):

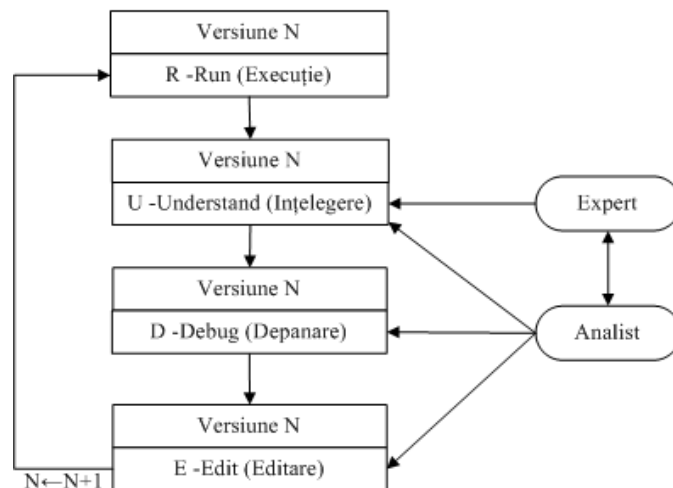


Fig.37 Metodologia Rude pentru sisteme inteligente

În cadrul acestei metodologii este necesară o metodă de construire a unui sistem inițial, de la care să pornească iterarea.

Din metodologia RUDE s-a dezvoltat o altă metodologie denumită POLITE (Produce Objective-Logical/Physical Design Implement-Test-Edit) Aceasta se poate folosi atât pentru dezvoltarea de sisteme clasice cât și pentru sisteme inteligente.

4.3.2.4. Etapele realizării unui sistem expert folosind prototipizarea

De la început, dezvoltarea sistemelor expert a fost asociată cu prototipizarea. În practică există cazuri în care sistemele sunt etichetate “sisteme bazate pe cunoștințe” doar pentru a putea utiliza o metodologie de dezvoltare mai puțin formală.

Conform lui Jojo și O’Keefe¹⁰³, există trei forme de prototipizare:

- *prototipizarea rapidă* utilizată de regulă când se lansează timpuriu diverse prototipuri ale sistemului, dar care au o durată de viață scurtă fiind înlocuite de versiuni mai noi
- *prototipizarea incrementală* în care sistemul este divizat în părți, fiecare fiind prototipizată și apoi inclusă una câte una în sistemul final
- *prototipizarea evolutivă* în care sistemul este rafinat mereu până satisface cerințele utilizatorilor.

¹⁰³ [Jojo94] Jojo L., O’Keefe R.M “Experiences with an expert system prototyping methodology” în Expert Systems: The International Journal of Knowledge Engineering, 1994.

Bazându-ne pe aceasta, procesul de elaborare a unui sistem expert poate fi descompus în patru etape¹⁰⁴:

1. *Studiul de fezabilitate*
2. *Proiectarea prototipului*
3. *Realizarea prototipului*
4. *Realizarea sistemului expert pornind de la prototip.*

1. Studiul de fezabilitate este un instrument care constituie suportul în decizia de a realiza respectivul sistem expert. Această investiție va fi făcută dacă se previzionează că sistemul își va atinge obiectivele din punct de vedere economic în perioada sa de viață, putând fi considerat astfel o investiție corectă. Studiul de fezabilitate trebuie să poată răspunde la următoarele întrebări:

- ✓ sistemul expert este realizabil?
- ✓ în ce măsură va ameliora situația existentă?
- ✓ care sunt persoanele capabile să-l realizeze?
- ✓ care este bugetul previzionat care se poate aloca?
- ✓ în cât timp se poate realiza?
- ✓ care sunt mijloacele necesare pentru punerea în funcțiune?
- ✓ care va fi rentabilitatea?

Există un set de cerințe la care o problemă sau lucrare care urmează să fie abordată prin tehnologia sistemelor expert trebuie să răspundă și anume:

- trebuie să facă apel la cunoștințe aparținând unui domeniu bine delimitat
- să fie structurabilă, adică să utilizeze elemente identificabile, reproductibile și formalizabile, atât în privința conceptelor și cunoștințelor folosite cât și a raționamentelor
- să reprezinte o lucrare repetitivă în timp
- să fie bine cunoscută de experții în domeniu
- să nu necesite tipuri de cunoștințe care nu pot fi reprezentate prin una din metodele de reprezentare cunoscute

Selectarea unei probleme care răspunde acestor criterii constituie numai un prim pas, care trebuie continuat cu o analiză mai detaliată, menită să verifice oportunitatea rezolvării sale printr-un sistem expert. În acest scop se pot avea în vedere patru categorii de factori:

- natura problemei sau a activității
- rezultatele previzibile
- expertiza disponibilă
- utilizatorii finali

a) *Din punct de vedere al naturii problemei*, evaluarea se bazează pe măsura în care aceasta răspunde următoarelor criterii:

- lucrarea este cognitivă, solicitând într-o măsură preponderentă analiză, sinteză, decizie
- implică raționamente sau cunoștințe simbolice
- face apel la un număr ridicat de parametri
- folosește reguli și metode euristice
- se bazează deseori pe date incomplete
- solicită oferirea de explicații și justificări ale rezultatelor și raționamentelor
- se află pe un nivel intermediar de formalizare, caracterizat prin preponderența clasificărilor și a euristicilor
- implică înlănțuiri și raționamente, structurate eventual pe mai multe niveluri
- nu poate fi rezolvată cu metode informatice convenționale.

b) *Din punct de vedere al rezultatelor*, evaluarea se bazează pe raportarea efectelor previzibile ale sistemului la următoarele criterii:

- creșterea cifrei de afaceri
- reducerea costurilor
- îmbunătățirea calității produselor sau serviciilor

¹⁰⁴ [Cârstoiu94] Cârstoiu D.I. –Sisteme expert, Editura All, București, 1994 (adaptare).

- colectarea expertizei pentru care nu există documentație scrisă
- asigurarea și difuzarea expertizei către personalul lipsit de experiență
- utilizarea, antrenarea și creșterea calificării profesionale

c) *Din punct de vedere al expertizei disponibile*, se urmăresc următoarele criterii:

- dacă există unul sau mai mulți experți recunoscuți în domeniu
- performanțele expertului în rezolvarea problemei sunt net superioare în raport cu cele ce pot fi obținute de către alte persoane
- dacă experții sunt disponibili pe o perioadă de timp suficientă
- dacă experții sunt de acord să elaboreze cazuri pentru testarea sistemului și să participe la evaluarea lui

d) *Din punct de vedere al viitorilor utilizatori*, se iau în vedere următoarele aspecte:

- dacă utilizatorii simt necesitatea sistemului
- dacă formulează cerințe și nu așteaptă servicii nerealiste din partea sistemului
- dacă au un anumit nivel de cunoștințe, un sistem expert neputând fi utilizat de persoane fără anumite cunoștințe profesionale corespunzătoare.

Răspunsurile la aceste întrebări sunt date de către persoanele specializate. Studiul de fezabilitate permite verificarea rapidă dacă pentru problema dată se justifică realizarea sistemului expert.

O tehnică utilizată frecvent în efectuarea unor asemenea analize o reprezintă grilele de evaluare (*checklist*). Acestea sunt compuse dintr-un ansamblu de criterii predefinite, în raport cu care problema abordată este apreciată cu un anumit punctaj. Problema este considerată oportună doar dacă punctajul cumulat depășește o anumită valoare.

2. Proiectarea prototipului

În cadrul acestei etape se stabilesc:

- sursele și tipurile cunoștințelor;
- modul de reprezentare și de utilizare a cunoștințelor;
- interfețele necesare;
- alegerea instrumentelor hardware și software.

În privința transferului de cunoștințe, aceasta se realizează prin identificarea și reținerea cunoștințelor din domeniu. Aceste cunoștințe pot fi fapte, ipoteze sau reguli și vor fi transferate în baza de cunoștințe a prototipului.

În privința alegerii mijloacelor informatice necesare, există două variante:

- elaborarea unui motor de inferențe propriu, scris într-un limbaj adecvat (în general Prolog sau Lisp)
- utilizarea unui generator de sisteme expert, în care va trebui doar creată baza de cunoștințe.

Alegerea între aceste două variante trebuie să aibă în vedere următoarele aspecte:

- limbajul de reprezentare al cunoștințelor să fie simplu și cât mai general
- să existe un mecanism de control eficace și un modul explicativ elaborat
- interfața cu utilizatorul să se apropie cât mai mult de limbajul natural

3. Realizarea prototipului -La finalul studiului de fezabilitate, problema trebuie să fie clar definită, permițând să se ia o decizie asupra desfășurării ulterioare a lucrării. Soluții distincte ale problemei trebuie testate și comparate pe parcursul realizării prototipului (care poate fi o extensie a machetei). În principiu prototipul trebuie să fie abandonat în momentul în care inginerul de cunoștințe are o idee precisă asupra dezvoltărilor viitoare ale sistemului expert pe baza experiențelor realizate. Prototipul obținut servește doar ca suport de decizie pentru autorizarea trecerii la faza următoare.

În cadrul acestei etape pot exista mai multe versiuni de prototip. Astfel, va exista un *prototip inițial*, constituind versiune operațională inițială a sistemului. Ulterior, prin rafinări și detalieri succesive, se obțin *prototipuri intermediare*, a căror număr poate diferi în funcție de complexitatea sistemului. După finalizarea fiecărei versiuni de prototip, se face o evaluare a acestuia, iar în momentul în care acesta va satisface exigențele cerute, acesta va constitui *prototipul final*, acesta din urmă devenind baza de la care se va dezvolta sistemul expert.

4. Realizarea sistemului expert începe prin determinarea utilitatelor corespunzătoare (ce pot fi aceleași cu cele ale fazei a 2-a). În etapa de dezvoltare se procedează la o rescriere a prototipului testat în etapa anterioară urmărind satisfacerea cerințelor de eficacitate. Baza de cunoștințe a sistemului expert se mărește lent pe măsura avansării proiectului, prin noi cunoștințe și prin rezultatele testelor ce demonstrează validitatea coerenței cunoștințelor. Ea va încorpora cazuri reale și din ce în ce mai complexe, iar sistemul expert începe să devină utilizabil înainte ca baza de cunoștințe și obiectivul fixat inițial să fie realizat complet. Trebuie amintit că faza de extracție a cunoștințelor este lungă și delicată, ea consumând o mare parte din timpul total de realizare a sistemului expert. După validarea bazei de cunoștințe și prezentarea versiunii operaționale are loc implementarea sistemului.

Dezvoltarea într-o variantă clasică ar presupune descompunerea problemei în subprobleme realizate independent unele de altele. În acest caz, abordarea corectă este cea experimentală și incrementală a machetei și a prototipului. Construcția printr-un proces experimental și incremental se bazează însă pe luarea în considerație la anumite etape a tuturor subproblemelor într-o manieră detaliată.

Multe companii se găsesc azi la sfârșitul etapei întâi, având o machetă realizată fără însă ca problemele să fie clar delimitate. Dacă decizia de trecere la etapa a 2-a se ia, partea cea mai importantă acum începe. Ca efect, un studiu asupra domeniului este absolut necesar și indispensabil. El permite determinarea cunoștințelor utilizate de expertul uman și maniera de a le reprezenta, sprijinindu-se pe experiența deja acumulată în acest domeniu (faza de conceptualizare, formalizare). Această etapă este importantă căci ea condiționează urmarea studiului. Lucrările care sunt efectuate (reprezentarea cunoștințelor, utilitate, mecanism de inferență) determină caracteristicile sistemului expert viitor și performanțele lui.

4.3.2.5. Ciclul de viață al unui sistem expert

Orice sistem expert parcurge mai multe etape de la analiză până la exploatare, activitatea sa încheindu-se odată cu scoaterea din funcțiune. *Modelul liniar* este un model utilizat cu succes la dezvoltarea unui număr mare de proiecte de sisteme expert, fiind adaptat după Bochler în 1988. Conform acestui model, majoritatea autorilor, printre care și Zaharie¹⁰⁵, prezintă ciclul de viață al unui sistem expert ca fiind compus din următoarele etape:

- **analiza preliminară** – această etapă presupune o analiză preliminară a activității care se dorește a fi modelată cu ajutorul sistemului expert cât și alegerea instrumentelor software necesare proiectării sistemului. Se consideră că o problemă poate face obiectul unui sistem expert numai dacă expertul uman poate să-i expună complet rezolvarea prin telefon (prin așa-numitul “test al telefonului”). În elaborarea unui sistem expert trebuie să se răspundă la diferite întrebări care vor stabili domeniul de interes, persoanele implicate în luarea deciziei, persoanele care vor construi sistemul, utilizatorii, scopul sistemului etc. Una din cele mai importante analize este aceea care trebuie să precizeze garanția atașată răspunsurilor furnizate de către sistem cât și eficiența acestuia. De asemenea trebuie analizat dacă sistemul este destinat să înlocuiască operatorii umani sau doar îi va asista în rezolvarea sarcinilor. Se va decide de asemenea dacă sistemul se va exploata într-o manieră liberă de către persoanele care au nevoie la un moment dat de expertiza sa, sau va fi utilizat de către persoane însărcinate special în acest sens, dacă va fi folosit permanent sau periodic. Sistemul poate fi folosit de experți în vederea ameliorării performanțelor sau de către nespecialiști în mod ghidat, sau poate fi destinat unor utilizatori multipli, funcționând pe diverse terminale. Dacă va trebui să funcționeze în timp real, trebuie prevăzute caracteristicile suplimentare ale calculatorului. De asemenea, trebuie prevăzute legăturile cu alte module sau sisteme (baze de date, algoritmi de calcul, generatoare de rapoarte, prelucrări ulterioare ale rezultatelor). Analiza trebuie să stabilească unde va fi utilizat sistemul, dacă acesta reprezintă o aplicație dedicată unei firme sau unui domeniu restrâns sau dacă va fi utilizat într-un număr mare de exemplare, stabilindu-se astfel generalitatea sistemului. Tot în această fază trebuie abordată problema costului sistemului. Calculul va include costurile dezvoltării sistemului, achiziția și introducerea cunoștințelor, costuri hardware etc. Se vor estima și beneficiile în termenii productivității, securității sporite, ameliorarea condițiilor de lucru, creșterea calității

¹⁰⁵ [Zaharie93] Zaharie D., Năstase P. –Sisteme expert de gestiune, Editura Romcart, București, 1993.

produselor și a serviciilor, diminuarea termenului de răspuns, o mai bună desfășurare a muncii, câștigul de timp.

- modelarea conceptuală - urmărește definirea structurii conceptuale a cunoștințelor utilizate de către expertul uman în domeniul de activitate care face obiectul viitorului sistem expert. Se definesc noțiunile de bază folosite, relațiile esențiale între aceste noțiuni, operațiile principale de efectuat și restricțiile generale care intervin în cursul rezolvării problemelor.
- colectarea cunoștințelor - urmărește colectarea tuturor cunoștințelor necesare funcționării viitorului sistem expert. Cunoștințele colectate în cadrul acestei etape nu sunt direct utilizabile; ele urmează a fi structurate și exprimate în forma adecvată metodei de reprezentare a cunoștințelor și a particularităților motorului inferențial.
- reprezentarea cunoștințelor - urmărește asigurarea formalizării și reprezentării cunoștințelor în structura adecvată înregistrării lor în baza de cunoștințe și exploatarei de către motorul inferențial.
- verificarea și validarea sistemului - verificarea urmărește dacă sistemul funcționează corect iar validarea cercetează dacă acesta îndeplinește cerințele propuse. Verificarea este în mod tipic o problemă de programare și își propune eliminarea erorilor din sistem, a conflictelor logice, a redundanței și deficiențelor. Conflictelor sunt concluzii inconsistente având la bază cunoștințe consistente. Redundanțele se referă la prezenta unor structuri logice care nu sunt necesare, neafectând legăturile dintre intrările și ieșirile sistemului. Deficiențele reprezintă absența unor structuri logice care ar trebui să fie prezente în sistem pentru a ajunge la concluzii pentru toate cazurile de intrări valide. Validarea urmărește calitatea sistemului și în ce măsură sistemul obținut poate fi utilizat în practică. Implicit, verificarea face parte din validare. În funcție de rezultatele observate, validarea poate impune completarea, rafinarea sau îmbogățirea bazei de cunoștințe ca și operarea de modificări structurale menite să îmbunătățească parametri de exploatare curentă. În general validarea urmărește atingerea unui nivel acceptabil de performanță al sistemului
- introducerea în exploatare și menținerea în funcțiune - cuprinde ansamblul activităților necesare trecerii la stadiul de sistem operațional precum și asigurarea evoluției sistemului prin completarea sau modificarea ulterioară a bazei de cunoștințe. Se apreciază că acestor activități le revine până la 40-50% din costul total al sistemului¹⁰⁶.

¹⁰⁶ [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.

În figura 38 este evidențiată succesiunea etapelor în cadrul ciclului de viață al unui sistem expert:

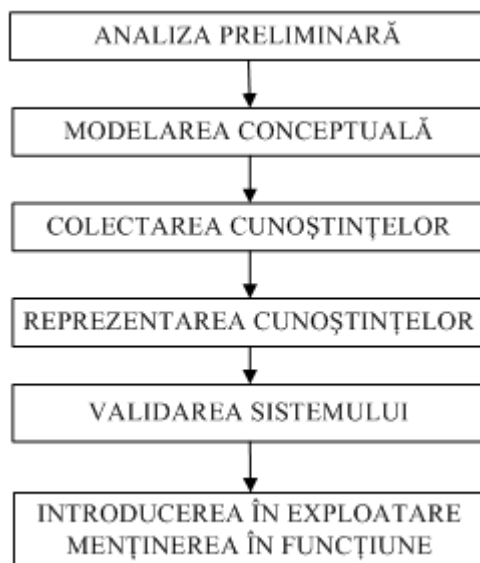


Fig. 38 Modelul liniar al structurii ciclului de viață al unui sistem expert

Considerăm că procesul de dezvoltare a sistemelor expert nu este liniar, deoarece unele faze se pot executa simultan sau pot avea numeroase reveniri la fazele anterioare. De altfel, metodologiile de realizare a sistemelor expert se bazează pe *paradigma realizării evolutive* a sistemelor software, care diferă de modelul liniar, al trecerii o singură dată printr-o serie de etape, faze, sau activități etc. Conținutul diferitelor etape și modul în care este dirijată reluarea acestora reprezintă elementele specifice fiecărei metodologii în parte.

Conform acestei paradigme se definește *modelul în spirală al ciclului de viață* al unui sistem expert (figura 39):

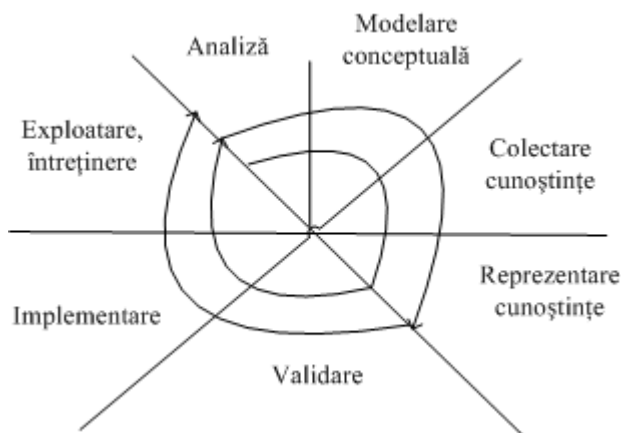


Fig.39 Realizarea evolutivă a sistemelor expert

Prin realizarea evolutivă a sistemelor se realizează o dezvoltare incrementală a bazei de cunoștințe, proces care este favorizat și de caracterul declarativ al majorității modalităților de reprezentare a cunoștințelor. Studiile arată că ritmul de achiziție al cunoștințelor se accelerează după construirea primei versiuni a sistemului.

4.3.2.6. Utilitatea și limitările sistemelor expert

Aplicațiile timpurii de calcul foloseau limbaje destinate manipulării valorilor numerice și erau potrivite pentru calcule ingineresti, scopuri științifice sau economico-financiare. Prin dezvoltare, tehnica de calcul a devenit mai ieftină și mai eficace, au apărut procesoarele de texte și sistemele de gestiune a

bazelor de date. Următoarea etapă de dezvoltare a fost caracterizată de înmagazinarea, interpretarea și obținerea cunoștințelor, ce s-a materializat sub forma sistemelor expert.

Utilizatorii pachetelor de programe tradiționale, de regulă, nu au cunoștințe despre programele folosite (dacă acestea au fost realizate de către altcineva). Numai autorii pachetului înțeleg funcționarea programului, respectiv ipotezele fundamentale. Sistemele expert sunt ceva mai transparente, ele îngăduind de regulă utilizatorului înțelegerea funcționării interne a sistemului.

Cunoașterea principiilor fundamentale și posibilitatea obținerii explicațiilor determină caracterul comprehensibil al sistemului pentru utilizator, de aici derivând și încrederea mai mare a utilizatorului în sistem. Deoarece aceste sisteme sunt complet interactive (accentuând la ele caracterul cu adevărat “prietenos”), ele reprezintă instrumente ideale pentru realizarea sistemelor de învățare, instruire, cu ajutorul cărora cel care învață își poate însuși cunoștințele fără a fi încurcat, amenințat sau fără a avea impresia că este condus de o “cutie neagră”.

Programele tradiționale de aplicații erau create în urma unei analize efectuate de către un specialist, analiză urmată de proiectarea de către un alt specialist și programarea de către un programator –utilizatorul doar folosindu-l. Acest proces părea să fie eficient, dar în realitate îngreuna destul de mult viața. În aceste sisteme erau atât de multe relații, conexiuni, încât utilizatorul nu putea fi un partener adevărat în procesul de dezvoltare. Orice schimbare a sistemului presupunea rescrierea unei părți considerabile din program iar aceasta influența și alte părți din sistem.

Comparativ cu acest mod de lucru, sistemele expert sunt mai flexibile, cu mai multe posibilități. Ele pot porni cu un domeniu mic, special, putând crește în timp nu numai sub aspectul extinderii bazelor de cunoștințe dar și prin posibilitatea extinderii acestora. Deoarece cunoașterea este un set fără limite de reguli, extinderea este nelimitată practic. Mașina de inferență lucrează cu din ce în ce mai multe reguli, deoarece orice s-ar întâmpla cu regulile ea rămâne neschimbată iar programul nu trebuie rescris.

4.3.2.8. Impactul utilizării sistemelor expert asupra activității unei firme

Principalele aspecte referitoare la impactul utilizării sistemelor expert asupra activității unei firme se pot defini astfel¹⁰⁷:

Impactul asupra productivității

Evaluarea câștigului de productivitate depinde de natura sarcinii de îndeplinit de către sistem. Măsurarea productivității se poate face prin procedee de simulare, înainte de instalarea definitivă a sistemului. De exemplu, în cazul unui sistem de diagnostic se câștigă timp, în condițiile proceselor în timp real și în gestiunea sistemelor de alarmă se obține vigilență de moment și siguranța reacției.

Productivitatea proceselor informaționale și decizionale este întotdeauna greu de evaluat. În cazul în care un sistem expert achiziționează o mare cantitate de cunoștințe pe care puțini experți le posedă și care este pus la dispoziția unei rețele de utilizatori, se obține un câștig de productivitate evidentă.

În mod obișnuit, câștigul de productivitate se măsoară prin economia de manoperă. Comparăția între situația anterioară introducerii sistemului și cea ulterioară nu este întotdeauna relevantă deoarece se profită de introducerea sistemului informatic pentru a adăuga noi lucrări.

Impactul asupra organizației

Este posibil ca organizația și persoanele implicate să fie refractare la apariția sistemelor expert, deoarece el poate perturba situația persoanelor implicate. În acest caz, considerăm că trebuie avut în vedere mediul și locul sistemului expert în organizație și în același timp găsiți utilizatorii care să se folosească de sistem și care îl înțeleg (rolul interfețelor), după care trebuie să se constituie grupul însoțitor a cărui sarcină este de a actualiza permanent cunoștințele din domeniul respectiv. Dacă baza de cunoștințe se îmbogățește constant, această acumulare devine un mare câștig pentru orice întreprindere. Sistemele

¹⁰⁷ [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.

expert pot fi sursa de inspirație pentru strategiile și orientările sale, contribuind astfel la dezvoltarea organizației și la bunul mers al activității acesteia.

Ca urmare, apreciem că deciziile luate vor orienta firma spre produsele cele mai rentabile sau vor determina crearea de noi produse sau servicii. În domeniul gestiunii produselor și prospectării pieței, sistemul poate orienta întreprinderea spre piețele și prețurile cele mai avantajoase, efectele traducându-se prin creșterea cifrei de afaceri.

Probleme umane

Impactul pe care îl aduce odată cu sine sistemul expert asupra întreprinderii creează în același timp și unele probleme umane. Acestea pot fi de trei feluri: probleme legate de introducerea unei tehnologii noi, probleme legate de schimbări de calificare, de creări/dispariții de locuri de muncă și probleme legate de experți.

Introducerea unui sistem informatic necesită, întotdeauna, o reorganizare a serviciilor, fiind necesară o redistribuire a sarcinilor între diferiții parteneri, ceea ce atrage după sine definirea noilor metode de lucru. Reorganizarea are ca efect, de cele mai multe ori, creșterea ponderii muncii de analiză și concepție, în detrimentul muncii de rutină.

Construirea bazei de cunoștințe, prin formalizarea expertizei, va duce la creșterea fezabilității expertizei și a puterii de raționament a expertului, pe care poate acesta nu a avut-o până în prezent și se constituie astfel într-o bună ocazie pentru expert ca să-și îmbunătățească performanțele. S-a constatat că ingineria cunoștințelor îi permite expertului să-și pună întrebări și să înțeleagă mai bine expertiza și este un nou pas spre progres pentru el. Experții înțeleg că această bază de cunoștințe le dă posibilitatea să lucreze mult mai bine și mai repede ca înainte. Expertul va face parte din grupul de întreținere al sistemului, deoarece el cunoaște cel mai bine baza de cunoștințe și el este în stare să-i mențină coerența după fiecare modificare.

Avantajele unui sistem expert variază în funcție de tipul sistemului precum și de domeniile în care se utilizează. Principalele avantaje ale unui sistem expert sunt următoarele¹⁰⁸:

-*Depozitarea expertizei* – baza de cunoștințe creată în urma implementării unui sistem expert reprezintă una din cele mai importante resurse ale întreprinderii. Acesta va putea fi folosită fără întrerupere, fără pericolele specifice ale activității umane (greve, concedii de boală și de odihnă). Totuși, fără actualizarea periodică a acestei baze de cunoștințe ea se va perima, devenind depășită.

-*Automatizarea și îmbunătățirea deciziilor* – studiile efectuate apreciază că 80% din munca unui expert uman se poate automatiza. Este vorba atât e operațiile simple, repetitive, procedurale dar și de operațiile mari consumatoare de timp și energie pe care sistemele expert le realizează ușor, rapid și eficace. Uneori și expertul poate greși, datorită unor diverse cauze (oboseală, neluare în considerare a tuturor ipotezelor etc). Acestea nu se aplică la deciziile unui sistem expert, care va raționa mereu la fel (bine înțeles, ținând cont și de modificările care apar în baza de cunoștințe).

-*Difuzarea expertizei și normalizarea deciziilor* – Implementarea unui sistem expert permite folosirea unui personal mai puțin calificat (și în consecință mai ieftin). De exemplu, sistemul expert instalat într-o centrală bancară își poate difuza expertiza pentru soluționarea unor probleme care intervin la ghișeele de lucru cu publicul, unde în general personalul este mai puțin calificat. Se urmărește astfel o răspândire uniformă a raționamentelor sistemului expert, fapt care îl face să reziste la preferințe sau aversiune față de clienți.

-*Învățarea și formarea continuă a utilizatorilor* – Sistemul expert este un excelent pedagog pentru toate tipurile de utilizatori, mai ales datorită facilității de explicare a raționamentelor. Chiar și expertul își poate verifica expertiza prin comparație cu soluțiile oferite de sistem.

-*Potențialul comercial* – Piața sistemelor expert este deja bine dezvoltată. Firmele au conștientizat importanța implementării și utilizării unor astfel de sisteme. Deținerea unor sisteme expert ridică valoarea unui întreprinderi, fapt care se poate vedea mai ales prin creșterea prețului acțiunilor respectivei întreprinderi la bursa de valori.

De asemenea avantajele se văd și din prisma reducerii costurilor, a îmbunătățirii calității produselor sau serviciilor, îmbunătățirea calității actului managerial etc.

O întrebare care se pune este de ce sistemele expert nu s-au răspândit într-un cerc mai larg? Motivul principal se bazează pe faptul că orice rezolvare își are limitele, nefăcând excepție de la această afirmație nici sistemele expert. S-a afirmat în multe rânduri că sistemul expert este o aplicație care caută probleme. Considerăm că există o doză de adevăr în această afirmație, mai ales dacă recunoaștem că sistemele expert s-au dezvoltat în mare măsură pe baza aplicațiilor inteligenței artificiale și mai puțin pe rezolvarea unor probleme reale.

¹⁰⁸ [Andone99] Andone I. Țugui A. –Sisteme inteligente în management, contabilitate, finanțe-bănci, Editura Economică, București, 1999.

Dezvoltarea oricărui sistem necesită un timp considerabil, dar acest timp este scurt în comparație cu durata utilizării. Sistemul este practic oglindirea cunoștințelor unei anumite perioade. Dacă baza de cunoștințe este bine organizată și documentată corespunzător, atunci ea poate fi actualizată, extinsă dar modelul va rămâne același pe toată durata de folosință a sistemului. Fiecare bază de cunoștințe trebuie concepută astfel încât să se acomodeze mediului de generare pentru sisteme expert și presupune o viziune proprie asupra lumii. Această viziune poate fi corectă azi dar mâine se poate dovedi incorectă.

Folosirea unui singur expert la realizarea setului de reguli presupune cunoașterea tuturor regulilor de către acel expert. Una din problemele care se pun este că un sistem expert, oricât de bine ar fi făcut, nu poate stoca toate cunoștințele, în principal din cauza dificultății enunțării celor ce țin de experiență, de intuiție. Rezultă că un sistem expert nu poate fi niciodată mai bun ca un expert uman de vârf. Deși sistemele expert nu pot duplica în întregime cunoștințele unui expert uman de vârf, totuși ele adaugă cunoștințelor expertului toată puterea de procesare care o posedă un calculator. Astfel, se pot face analize mult mai rapid, se pot prelucra cantități uriașe de date, se poate lucra cu situații deosebit de complicate. De aceea, în unele cazuri, sistemele expert pot îndeplini sarcini mult mai bine ca un expert uman. De exemplu, pot lua în considerare chiar și cele mai improbabile situații, pe când expertul uman se rezumă, în general, la o selecție a situațiilor cele mai probabile, bazându-se pe intuiția și experiența sa, datorită faptului că are o viteză de analiză redusă și o analiză completă îi ocupă foarte mult timp.

De asemenea, se presupune că ne-am clarificat asupra tuturor regulilor. Cu alte cuvinte, am putea declara că nu are importanță alegerea expertului uman, pentru că oricare ar fi el ar realiza același set de reguli. Acest punct de vedere este însă eronat. Probabil există o înțelegere largă între experții umani, dar ajungerea la un numitor comun (obținerea înțelegerii complete) între ei este foarte dificilă, aproape imposibilă. Cercetătorii au recunoscut problema unificării părerilor divergente și au dezvoltat diverse procedee tehnice pentru unificarea cunoștințelor experte.

De exemplu prin anii '70 a apărut conceptul procedurii "tabla neagră" (*blackboard*) în care se inventariază toate opiniile divergente, încercând găsirea unor relații comune înainte de a le cuprinde și pe acestea în baza de cunoștințe. Dezvoltarea și perfecționarea acestui concept continuă și astăzi. Practic o asemenea tablă neagră a ajuns o bază mare și compusă de date, care ține evidența conținutului bazelor de cunoștințe ale domeniilor parțiale ce formează domeniul dat. Pentru tratarea complexității caracteristice acestui procedeu sunt însă necesare calculatoare ce suportă prelucrări paralele corespunzătoare.

Noțiunea de sistem expert presupune implicit existența unei soluții optime sau cel puțin viabile. Se pot imagina însă unele probleme care sunt nerezolvabile și fără răspuns la nivelul actual al cunoștințelor noastre. În aceste cazuri sistemele expert nu au cum să ne furnizeze răspunsuri. Se poate întâmpla ca răspunsul furnizat de sistemul expert să nu fie utilizabil sau acesta să fie utilizabil însă considerat inacceptabil de către utilizator. Ce se întâmplă în acest caz cu credibilitatea sistemului? Fiecare sistem trebuie să se poată adapta elastic la viziunea proprie unică a utilizatorului. Sistemele expert presupun existența unei lumi perfecte, însă aceasta nu există încă. De aceea considerăm că scopul este obținerea în locul unor soluții optime a unor soluții acceptabile, pentru că lucrăm cu resurse limitate. O soluție (chiar optimă) are puține șanse să fie la fel de bună pentru toți. În general nu există timp, bani, informații, instrumente sau forță de lucru suficiente pentru obținerea soluțiilor perfecte.

4.3.3. Generatoare de sisteme expert

4.3.3.1. Medii și generatoare pentru sistemele expert

Un sistem expert –oricât ar fi de performant- nu este numai un motor de inferență, ci necesită mai ales o acumulare masivă de cunoștințe care trebuie introduse și gestionate în sistem. De asemenea trebuie interpretate și înțelese rezultatele oferite de sistem. Toate aceste funcțiuni se găsesc în afara atribuțiilor și posibilităților motorului inferențial, ele fiind realizate cu instrumente auxiliare numite *utilitare*. Noțiunea de sistem expert se poate întâlni frecvent și referitor la aceste instrumente (care sunt pachete de programe fără baze de cunoștințe dar care sunt capabile să primească și să exploateze cunoștințe organizate într-o bază). Aceste instrumente se numesc *generatoare de sisteme expert* (GSE) sau medii (din traducerea termenului englezesc *shell*). Un asemenea generator conține, în afara motorului de inferență, utilitare de elaborare și exploatare necesare.

Un generator de sisteme expert reprezintă un produs program complex care integrează ansamblul cunoștințelor necesare elaborării și consultării sistemelor expert. În literatura de specialitate le întâlnim sub denumirea de sisteme esențiale¹⁰⁹: “toate componentele unui sistem expert, mai puțin baza de cunoștințe, formează așa-numitul sistem esențial”. Generatoarele sunt deci instrumente în care utilizatorul, cognitivian sau expert, furnizează numai baza de cunoștințe, celelalte componente fiind asigurate de generatorul însuși. În centrul unui GSE se află motorul de inferență care este conectat la baza de cunoștințe. Baza de cunoștințe trebuie încărcată și alimentată cu fapte și reguli, această operație fiind realizată cu ajutorul unui editor (toate sistemele evolute posedă un asemenea editor pentru achiziția cunoștințelor). Alături de acest modul indispensabil există și alte utilitare: trasorul, motorul de elaborare, modulul de învățare și/sau autoinstruire, eventual alte module profilate pe explicații, comentarii și calcule. Cele din urmă nu sunt prezente în toate sistemele expert, ele putând fi înlocuite de interfețe sau funcții specializate, după natura domeniului.

În concluzie, considerăm că un GSE satisface deci trei obiective:

- achiziționarea cunoștințelor experților
- exploatarea cunoștințelor prin motorul de inferențe
- menținerea și actualizarea cunoștințelor prin adăugare, modificare și ștergere

Fiecare modul al unui GSE poate fi echipat cu propria interfață de dialog. Pentru sistemele expert industriale, de regulă trebuie să existe și o interfață pentru achiziționarea automată a datelor de la senzori. În domeniul gestiunii, interfețele sunt destinate mai ales pentru extragerea datelor din baze și înregistrarea noilor informații.

Editorul, prin interfața care asigură și dialogul, trebuie să permită schimburi de cunoștințe (comunicare) într-un mod cât mai accesibil utilizatorilor potențiali (de exemplu într-un mod text cât mai apropiat de limbajul natural, în mod grafic utilizând semne și simboluri cât mai sugestive). Pentru ca utilizatorul să se facă înțeles de către calculator, este necesar să se familiarizeze și să se conformeze cu o sintaxă predefinită a comunicării. În măsura în care structura cunoștințelor permite, editorul poate verifica dacă faptele și regulile au forma cerută. De exemplu, în modulul de achiziție a cunoștințelor la sistemul MYCIN, editorul verifică dacă o regulă referitoare la identitatea unui organism are în premise următoarele trei elemente: tipul și locul infecției și modul de contaminare. Sistemele care posedă o structură semantică de genul rețea de cadre sau clase pot verifica coerența între rețea și datele furnizate de către utilizator. Dacă utilizatorul dorește să adauge obiecte sau reguli, editorul îl va chestiona referitor la modul de integrare al acestora în rețelele existente. Deci editorul poate controla dacă cunoștințele introduse corespund structurării predefinite a cunoștințelor. În schimb, el nu este capabil să detecteze toate tipurile de incoerențe. O mare parte dintre contradicții se pot elimina în faza de testare, în momentul manifestării acestora. Cercetarea aprofundată a incoerențelor ar putea fi rezolvată, eventual, de un modul special.

Editorul gestionează și dicționarul faptelor, obiectelor sau cadrelor, al regulilor (posedând lista completă a acestora) și confidențialitatea acceselor la bazele de cunoștințe. De multe ori poate fi utilă și asocierea unor comentarii la unele fapte și reguli, realizarea și ordonarea acestora fiind asigurată tot prin modulul de editare. În anumite sisteme, editorul își încheie sesiunea prin compilarea regulilor, ceea ce permite accelerarea activității inferențiale a motorului și detectarea erorilor de sintaxă din cadrul regulilor. Interfața editoarelor este de multe ori impresionantă, ca urmare a efortului imens depus de către proiectanții acestor sisteme.

Trasorul este un element esențial al sistemului expert, care joacă un rol important atât în modul de exploatare cât și în etapele de dezvoltare. Prima lui sarcină este urmărirea șirului raționamentelor care se efectuează prin motorul de inferență. Trasorul are de asemenea sarcina de a actualiza baza de fapte, pe măsură ce faptele sunt dovedite. În cadrul modulului de exploatare trebuie să fie posibilă citirea faptelor care sunt stabilite la un moment dat al sesiunii. Putem dori modificarea acestor fapte –la care trebuie să avem acces în cadrul sesiunii- mai ales dacă avem impresia că ne-am înșelat la un răspuns anterior. Când modificăm valoarea unui fapt, trasorul regăsește toate deducțiile care au fost făcute pornind de la acest fapt și le șterge. Sesiunea se poate relua cu noua valoare (și implicit noua stare). Asemenea încercări de modificare permit evaluarea sensibilității rezultatului final raportat la valoarea faptelor sigure (certe).

¹⁰⁹ [Zaharie93b] Zaharie D, Năstase P. –Sisteme expert de gestiune, Revista “Tribuna Economică” nr. 10/1993.

Această funcțiune, numită *studiu de sensibilitate*, este una din posibilitățile importante oferite utilizatorului de către generatoarele performante. Orice modificare a faptelor antrenează actualizarea imediată a bazei de fapte și continuarea consultării cu noile date. La sfârșitul sesiunii este disponibil ansamblul regulilor care au servit la demonstrarea unui scop, adică este accesibil traseul complet, precum și înlănțuirea raționamentelor.

Principalii factori în alegerea unui generator de sisteme expert se pot structura schematic conform figurii 40.

Avantajul principal al generatoarelor este acela că permit chiar și utilizatorilor neexperimentați să obțină rapid un sistem expert funcțional, iar prin rafinări succesive să ajungă la un sistem expert performant.

În concluzie, dezvoltarea unui sistem expert folosind unul sau mai multe medii și/sau instrumente de dezvoltare presupune adăugarea cunoașterii, reprezentând expertiza, obținută de la experți, depozitată în baza de cunoștințe și efectuarea testărilor și a rafinărilor necesare.

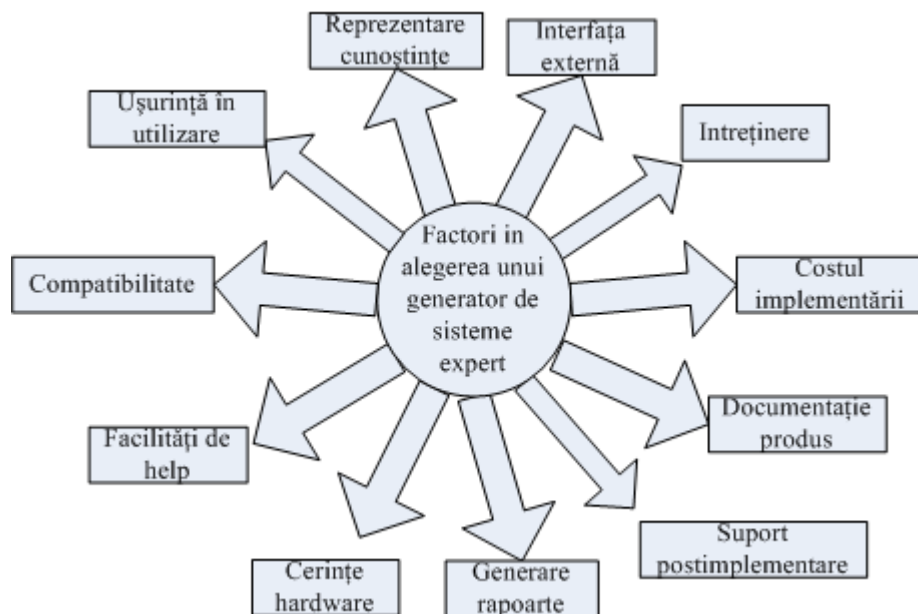


Fig. 40 Factori în alegerea unui generator de sisteme expert

Considerăm că un mediu și/sau instrument de dezvoltare reflectă un anumit punct de vedere asupra cunoașterii (reprezentare, organizare etc) și o metodologie specifică pentru construirea sistemelor expert. El orientează proiectantul către construirea unui sistem expert de tip diagnostic sau de alt tip, în conformitate cu paradigma implementată. O paradigmă specifică constituie o strategie pentru utilizarea cunoașterii în scopul soluționării unei clase de probleme. Fiecare paradigmă implică anumite proprietăți în proiectare pentru arhitectura sistemului expert, iar mediul și/sau instrumentul include aceste proprietăți în baza de cunoștințe și/sau motorul de inferențe.

4.3.3.2. Exemple de generatoare de sisteme expert

Tipologia generatoarelor de sisteme expert cuprinde:

- generatoare de dimensiune mică: EXSYS, VP-Expert, Level 5, CLIPS;
- generatoare de dimensiune medie: EXSYS-Developer, GURU, KES, N-expert, Personal Consultant Plus;
- generatoare de mare dimensiune: ESE-IBM, S.1, AES, IMPACT, SYNTEL, ADS, H-Expert.

Generatorul de sisteme expert GURU a fost dezvoltat de firma Micro Data Base Systems Incorporation. A fost gândit în ideea realizării, testării și implementării sistemelor expert de dimensiuni medii și mici. Pentru asigurarea unui număr mare de facilități, acest generator cuprinde:

- un limbaj specializat pentru specificarea elementelor necesare aplicațiilor (schema de reprezentare a cunoștințelor, stilul de realizare a interacțiunii cu utilizatorul);

- un editor pentru modificarea elementelor aplicației;
- utilitare pentru realizarea interfeței utilizator și a interfeței cu alte sisteme software, pentru întreținerea aplicațiilor, etc;

Acest generator de sisteme expert conține motorul de inferență, constructorul bazei de cunoștințe precum și un program de consultare a aplicației, cu generare de explicații atât în timpul consultării aplicației cât și la terminarea consultării. Generatorul GURU mai dispune și de o serie de componente ce permit funcții de prelucrare convenționale, respectiv:

- stocarea, gestionarea și prelucrarea eficientă a datelor prin tabele de calcul (spreadsheet-uri);
- generarea de grafice;
- comunicarea la distanță prin intermediul unui modem;
- execuția unor comenzi ale sistemului de operare fără a părăsi mediul.

GURU este deci un generator de sisteme expert ce îmbină avantajele instrumentelor software clasice (spreadsheet, baze de date, editor, interpretor, etc.) oferind numeroase facilități pentru dezvoltarea unor sisteme expert eficiente cu un efort acceptabil și într-un timp relativ scurt.

Generatorul de sisteme expert H – EXPERT reprezintă un instrument dezvoltat în C, deci cu o bună portabilitate pe diferite tipuri de mașini și destinat realizării de sisteme expert de dimensiuni mari.

Generatorul integrează concepte ale programării orientate obiect, utilizând conceptele de clasă, obiecte, moștenire multiplă, demoni, etc. Reprezentarea cunoștințelor se face prin intermediul regulilor de producție. El asigură o reprezentare ierarhică a bazei de cunoștințe, ceea ce face posibilă optimizarea timpilor de răspuns. Asigură interfețe cu baze de date, spreadsheet-uri, limbaje generale.

4.3.3.3. Etapele dezvoltării unui sistem expert cu ajutorul generatorului de sisteme expert

Dezvoltarea unui sistem expert cu ajutorul unui generator de sisteme expert implică parcurgerea succesivă a următoarelor etape:

- 1: Conceperea și implementarea generatorului de sisteme expert de către proiectant
- 2: Familiarizarea utilizatorului în privința folosirii generatorului de sisteme expert
- 3: Implementarea de către utilizator (sau utilizator și proiectant) a unei aplicații inițiale expert sau îmbunătățirea uneia existente
- 4: Folosirea efectivă a aplicației expert de către utilizator pentru rezolvarea problemei și adăugarea rezultatelor în catalogul de probleme rezolvate
- 5: Identificarea de către utilizator de noi probleme sau deficiențe în modul actual de rezolvare a problemelor după care se trece la pasul 3
- 6: Extinderea posibilităților de dezvoltare a generatorului de sisteme expert: adăugarea unui subsistem DBMS, extinderea conceptelor de modelare, extinderea posibilităților generatorului de rapoarte, adăugarea de noi sisteme după care se reia pasul 2.

Figura 41 ilustrează grafic succesiunea acestor etape.

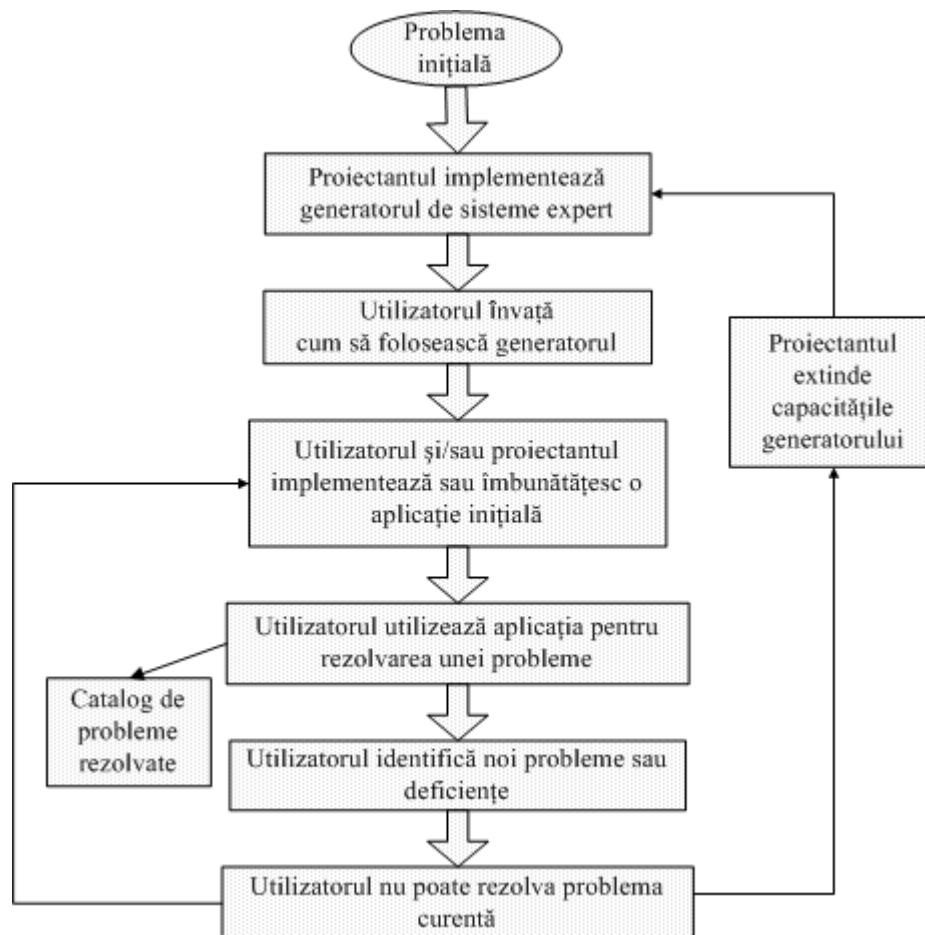


Fig. 41 Etapele dezvoltării unui sistem expert cu ajutorul generatorului de sisteme expert

4.3.4. Aplicații ale sistemelor expert în domeniul bursier

4.3.4.1. Prezentarea unor sisteme expert; structură și mod de utilizare

Printre sistemele expert cele mai cunoscute putem enumera:

ERS¹¹⁰ (Embedded Rule-Based System) permite nu numai realizarea unui sistem consultativ simplu, dar și a unui sistem expert ce poate funcționa ca o “interfață inteligentă”. Astfel, pentru evaluarea vârfurilor ce reprezintă faptele, ERS poate apela de la sine funcții ajutătoare specifice aplicației, fără a întreba utilizatorul despre faptele elementare. Aceste funcții pot furniza informații actuale din diverse baze de cunoștințe. În construirea sistemelor ce funcționează ca interfețe (pentru susținerea cărora a fost realizat ERS) se parcurg două faze esențiale. Prima dată trebuie realizată dezvoltarea bazei de cunoștințe –prin modul consultativ, după aceea trebuie dezvoltate funcțiile ajutătoare, pentru a cupla sistemul cu o bază existentă (sau cu un sistem decizional creat anterior). ERS a fost realizat în Pascal.

Sistemul decizional **DART** (Duplex Army Radio/Radar Targeting Aid), funcționând ca interfață inteligentă pe calculatoare VAX 11/80, are motorul de inferență foarte asemănător cu cel al lui ERS. Dezvoltarea bazei de cunoștințe a sistemului DART se începe prin folosirea în mod consultativ a sistemului vid. Acest lucru face posibilă încărcarea bazei de cunoștințe fără considerarea interfețelor

¹¹⁰ [***] ERS <http://www.ers.com>

necesare. Funcțiile ajutoare ale vârfurilor, ce reprezintă situațiile faptelor elementare, se conturează treptat, fiind din ce în ce mai bine definite pe măsura utilizării.

DELTA (Diesel Electric Locomotive Troubleshooting Aid) este un sistem expert realizat pentru a facilita depistarea defecțiunilor la locomotive și pentru furnizarea soluțiilor necesare remedierii. Sistemul dispune de un mecanism decizional hibrid: raționare prin avansare de la fapte la consecințe (înainte) și în direcție inversă, permițând urmărirea și în sens opus, adică deducerea faptelor din consecințe (înlănțuire înapoi). Modul general de gândire al sistemului se bazează pe paradigma eliminării erorilor, adică repetarea de un număr suficient de ori a ciclului de tipul: “Caută o greșeală, elimin-o, verifică dacă au mai rămas probleme”. Cunoștințele de specialitate sunt reprezentate prin reguli și/sau, forma simplificată a unei asemenea reguli fiind:

DACĂ motorul nu dezvoltă putere
 Și
 Nivelul presiunii combustibilului este scăzut
 Și
 Indicatorul de presiune a combustibilului este corespunzător
ATUNCI sistemul de alimentare cu combustibil este defect.

Partea condițională a regulilor este alcătuită din condițiile necesare și suficiente pentru depistarea defecțiunilor. Regulile sunt înmagazinate separat, conform apartenenței lor la diferite domenii de probleme. Pentru ca întotdeauna să se activeze doar regulile esențiale din punctul de vedere al mulțimii actuale de indici, la grupele de reguli se atașează indecși de către așa numitele metareguli. Sistemul DELTA folosește mai mult de 500 de reguli, din care aproximativ 300 servesc cunoștințele de diagnosticare iar restul sistemului de ajutor (help) –răspund întrebărilor puse de către utilizatori și oferă alte informații complementare. Deși original DELTA a fost conceput și realizat în LISP, a fost transcris ulterior în FORTRAN pentru a putea fi exploatat mai bine pe microcalculatoare. Rulează pe PDP 11/23 necesitând 10 Mb spațiu pe disc, cititor pentru discuri optice (CD-ROM) și terminal cu monitor grafic.

GEN-X (Generic Expert System) a fost realizat pentru sprijinirea dezvoltării sistemelor expert pe microcalculatoare. Dintre domeniile de aplicare se pot aminti: sisteme expert pentru descoperirea defectelor motoarelor la aeronave, sisteme expert pentru dirijarea proceselor industriale etc. GEN-X se compune dintr-un gestionar al bazei de cunoștințe, programe interpretatoare și generatoare de cod. Utilizatorul poate opta, alegând diverse forme de reprezentare a cunoștințelor (grafuri, arbori de decizie, etc). Gestionarul bazei de cunoștințe oferă instrumente grafice bazate pe dialog pentru realizarea, editarea și corectarea conținutului. Interpretatoarele conduc etapele consultative pe baza specificațiilor existente în baza de cunoștințe sau primite prin interfața utilizator. Sistemul generează separat baza de cunoștințe și sistemul deductiv, prin etapele consultative corespunzătoare. Dacă dezvoltarea bazei de cunoștințe atinge un nivel satisfăcător, atunci generatoarele de cod realizează traducerea în programe (folosind limbajele C, Ada, Fortran, Pascal). Deși GEN-X a fost realizat inițial pe calculatoare mai mari (în C), rulează și pe microcalculatoare, iar scopul declarat la realizarea lui a fost generarea codului pentru aplicații pe microcalculatoare. Pe un calculator personal (PC) sunt suficienți 32kb de memorie pentru o bază de cunoștințe ce conține 1000 de reguli de tipul și/sau.

APES (Augmented Prolog for Expert Systems), produs de Programming Logic Systems, este un mediu destinat realizării sistemelor expert care se bazează pe Micro-PROLOG, în care utilizatorul poate alege modul în care să fie tratate incertitudinile. Partea probabilistică a motorului de inferență poate fi combinată cu capacitatea deductivă logică implantată în PROLOG. Dintre domeniile în care există aplicații APES se enumeră: stabilirea dreptului la ajutor social, alegerea și clasificarea terenurilor pentru construirea barajelor, stabilirea tratamentului de urmat la anumiți pacienți incurabili, diagnosticarea corodării conductelor. Printre aceste aplicații există sisteme ce conțin mai mult de 400 de reguli. În prezent aproape fiecare sistem PROLOG are elaborat un mediu propriu (denumit tot APES) care poate fi folosit la dezvoltarea de sisteme expert.

ACLS¹¹¹ (Analog Concept Learning System) este în stare să genereze individual regulile de clasificare caracteristice domeniului abordat, din exemple diverse de aplicare. Față de sistemele anterioare, necesită o participare diferită din partea specialistului la elaborarea bazei de cunoștințe. La dezvoltarea unui sistem expert sub ACLS la început se definește mulțimea atributelor primitive și a claselor necesare organizării corespunzătoare a domeniului abordat. După această etapă, expertul uman comunică exemple evidențiatore sistemului (precizând valorile atributelor și regulile aparținătoare). Pe parcursul prelucrării acestora, ACLS induce individual regula generalizată pentru exemplele primite sub forma unui arbore de decizie. Atributele, comunicate de către specialistul uman, care se dovedesc mai târziu irelevante, nu ajung să fie inserate în regula generală. În final expertul uman poate evalua dacă regula indusă de către sistem reflectă sau nu corespunzător cunoștințele sale într-un mod suficient de exact. În cazul unui eșec, el poate efectua corecturi prin precizarea unor exemple și/sau proceduri noi.

ACLS poate funcționa atât sub formă de pachete de prelucrare (batch) cât și sub formă interactivă. În modul interactiv, utilizatorul poate verifica și examina pas cu pas arborele de decizie generat, urmărind modificarea acestuia după prelucrarea fiecărui exemplu. Sistemul poate genera și un cod executabil (folosind sintaxa limbajului Pascal), reprezentând arborele de decizie prin reguli decizionale structurate (if-then-else).

RuleMaster¹¹² –conține algoritmul inductiv de învățare prezentat la ACLS dar implantat într-un mediu mai extins de dezvoltare, fiind capabil în același timp să organizeze ierarhic arborii de decizie induși cu regulile, respectiv să prezinte explicații referitoare la deducțiile realizate în cadrul sistemului. Regulile induse sunt înmagazinate automat în module, prin intermediul limbajului Radial. Acest limbaj dispune de mecanisme pentru realizarea modulelor, verificarea și urmărirea aplicării regulilor, accesarea de resurse externe de informații, operatori probabilistici și fuzzy, module explicative. RuleMaster a fost realizat în C și poate rula sub sistemul de operare Unix pe calculatoare IBM PC, Sun Micro sau DEC Vax. Aplicabilitatea sistemului RuleMaster a fost demonstrată de două exemple practice. Sistemul Willard, folosit pentru presemnalizarea furtunilor puternice, este compus din 27 de module (fiecare din aceste module fiind un arbore de decizie indus în mod individual). Compararea sesizărilor furnizate de către acest sistem cu cele de la Centrul Național de Urmărire a Furtunilor SUA a demonstrat eficiența extraordinară a sistemului.

TIMM (The Intelligent Machine Model) conține un sistem de achiziționare de cunoștințe capabil să definească regulile automat, cuplat cu un motor de inferență fuzzy. Expertul uman definește la început problema, prin precizarea mulțimii de atribute și valorile posibile ale acestora. După această etapă, expertul cooperează cu sistemul pentru realizarea regulilor de producție: sistemul generează diferite situații pe baza definițiilor date pentru domeniul abordat, situații ce sunt evaluate de către specialistul uman. În etapa de consultare, sistemul va furniza utilizatorului curent răspunsuri și sugestii luând în considerare această evaluare (factorul decizional precizat de către expertul uman). Pe lângă această evaluare, expertul uman poate marca valorile atributelor ce caracterizează situațiile deosebite și poate fixa pentru acestea valoarea exactă a factorului decizional.

În TIMM este posibilă și examinarea globală a mulțimii de reguli. Sistemul poate depista situațiile insuficient acoperite de reguli prin valorile atributelor. Căutând regulile ce dispun de condiții asemănătoare, dar care au efecte contradictorii, sistemul poate efectua și o verificare a consistenței bazei de cunoștințe.

Aranjând atributele ce descriu condițiile regulilor (pe baza unor valori asociate) se obține un registru special în domeniul reprezentării cunoștințelor. Folosindu-se de acesta, motorul deductiv aranjează doar parțial condițiile regulilor, permițând ca partea condițională a unei reguli să nu se potrivească exact la o situație dată. Privit astfel, se poate afirma că mecanismul deductiv al sistemului TIMM este analogic, fiind apreciat că aplică întotdeauna regula cea mai potrivită situației date. Diversele

¹¹¹ [Paterson82] Paterson A. –ACLS User Manual, Intelligent Terminals Ltd., 15 Canal St. Oxford, OX2 6BH, UK, 1982.

¹¹² [Michie84] Michie D., Muggleton S., Riese C., Zubrick S. –RuleMaster: A Second Generation Knowledge Engineering Facility. Presented at the First Conference on Artificial Intelligence Applications, Denver, Colorado, 5-7 December 1984.

variante dezvoltate pentru IBM PC ale sistemului TIMM se ocupă cu preponderență de analize financiare și asigurări pentru ambarcațiuni. Sistemul TIMM a fost realizat în FORTRAN și poate fi rulat pe calculatoare compatibile IBM PC care dispun de coprocesor matematic încorporat sau pe sisteme mai mari. Pe IBM PC necesită 640kb RAM și un spațiu de 10Mb pe disc.

Personal Consultant era instrumentul conceput de către Texas Instrument, destinat proiectării și dezvoltării sistemelor expert pe calculatoarele profesionale TI. Acest sistem conține, în esență, proceduri alcătuite în principal din comenzi și este înzestrat cu structuri de reguli asemănătoare cu cele din EMYCIN. Personal Consultant a fost realizat în LISP. Configurația recomandată era alcătuită dintr-un TI Professional Computer cu sistem de operare MS-DOS (minim versiunea 2.1) dispunând de 512kb RAM și 10Mb spațiu pe disc. Într-o asemenea configurație permitea manipularea unui sistem alcătuit din 400 de reguli.

K:base oferă o altă modalitate de reprezentare a cunoștințelor față de sistemele prezentate, care funcționau mai mult sau mai puțin folosind aceleași scheme de reprezentare. Mai exact, pretindeau din partea expertului uman cunoștințe corespunzătoare euristicii folosite de către acesta, traduse în reguli individuale (separate) sau erau în măsură să inducă asemenea reguli, pe baza unor exemple furnizate de către experții umani.

Deși contemporan cu celelalte sisteme prezentate anterior, K:base reprezintă oarecum o cotitură față de acestea deoarece reprezentarea cunoștințelor în acest sistem s-a realizat prin cadre (frames) o structură apărută odată cu programarea orientată pe obiecte. Aceste cadre erau în esență structuri de date statice, care uneau cunoștințele declarative referitoare la obiecte cu cunoștințele procedurale despre folosirea și manevrarea acestora. Această abordare a reprezentării părea mai potrivită pentru bazele de cunoștințe complexe, în raport cu reprezentarea bazată pe reguli, simplificând semnificativ și reprezentările grafice pe terminale (monitoare) a cunoștințelor implementate în sistem. Inițial, domeniul de aplicare a sistemului K:base a fost analiza diferențelor dintre diverse rate ale dobânzilor. Sistemul a fost scris LISP pentru mașina LISP Symbolics 300 și pentru calculatoare IBM PC cu cel puțin 512 kb RAM.

CLIPS¹¹³ (C Language Integrated Production System) este unul din cele mai recente și mai celebre instrumente pentru dezvoltarea sistemelor expert. Acesta oferă la ora actuală un mediu complet pentru realizarea sistemelor bazate pe reguli și/sau obiecte. Este folosit pe scară largă mai ales în SUA, de la publicul interesat până la NASA, universități sau birouri federale. Prima versiune a acestui sistem a fost realizată, ca și prototip, în anul 1986 de către Secția de Inteligență Artificială a Centrului Spațial Johnson de la NASA. Sintaxa CLIPS a fost creată apropiat de sintaxa unui subset al sistemului expert ART dezvoltat de Corporația Inference.

Din punctul de vedere al reprezentării cunoștințelor, este o unealtă coezivă pentru tratarea unei varietăți extinse de cunoștințe cu suport pentru trei paradigme diferite de programare: pe bază de reguli, orientat obiect și procedural. Programarea pe bază de reguli permite reprezentarea cunoștințelor euristice prin specificarea unui set de reguli ce trebuie întreprinse pentru o situație dată. Programarea orientată obiect permite ca sistemele complexe să fie modelate prin componente modulare (care pot fi reutilizate ușor pentru a modela alte sisteme sau pentru a crea componente noi).

Posibilitățile oferite de CLIPS în domeniul programării procedurale sunt similare celor din limbajele C, Pascal, Ada și LISP. CLIPS a fost scris în C, pentru portabilitate și viteză de execuție și a fost implementat pe mai multe sisteme de calcul. A fost testat pe calculatoare compatibile IBM PC sub sistemele de operare DOS, Windows, pe Macintosh sub sistemele de operare MacOS și Mach. Având în vedere că CLIPS se livrează cu sursa completă (poate fi modificată și ajustată după nevoile utilizatorului), se poate implementa pe orice sistem de calcul care dispune de un compilator C. Versiunea standard a sistemului oferă un mediu interactiv de dezvoltare, cu unelte de depanare, help on-line și editor integrat. Sistemele expert CLIPS pot fi lansate în execuție în trei moduri: interactiv utilizând o interfață simplă cu prompter, interactiv utilizând o interfață cu ferestre și meniuri sau ca sistem expert înglobat. El se încadrează în paradigma limbajelor bazate pe reguli de producție. Pentru a rezolva o problemă în CLIPS

¹¹³ [***] CLIPS <http://www.siliconvalleyone.com/clips.htm>

programatorul definește baza de cunoștințe și alege strategia de căutare, iar sistemul dezvoltă un proces de aplicare a regulilor care se perpetuează atât timp cât mai există reguli de aplicat. Motorul de inferență cuprinde adăugare dinamică de reguli precum și strategii customizabile de rezolvare a conflictelor. Se poate ușor încapsula în alte aplicații. CLIPS include și un limbaj orientat obiect numit Cool (Clips Object Oriented Language) care este direct integrat cu motorul de inferență.

Există de asemenea facilități pentru verificarea și validarea sistemelor expert (permite proiectarea modulară și partiționarea bazei de cunoștințe, verificarea valorilor și argumentelor de funcții precum și a constrângerilor, analiza semnatică a seturilor de reguli pentru inconsistențe) și dispune de o documentație extinsă (ghid de utilizare, manual de referință, exemple). În prezent sistemul este menținut ca și software în domeniul public de către principalii autori (care nu mai sunt anagajați ai NASA) iar utilizarea acestuia în scopuri educaționale este gratuită. Există o listă de discuții ce privește această problemă la adresa:

listserv@cossack.cosmic.uga.edu

FuzzyClips este o versiune a generatorului de sisteme expert CLIPS cu extensii ce permit reprezentarea și manipularea faptelor și regulilor incerte. Sistemul poate combina faptele exacte cu cele incerte. Este disponibil pentru sisteme Unix, Macintosh și PC la adresa :

<http://ai.iit.nrc.ca/fuzzy/fuzzy.html>

WXClips oferă o interfață grafică sistemelor CLIPS. Folosește un stil de programare bazat pe evenimente și un set de funcții GUI. Este disponibil pentru transfer FTP la adresa:

<ftp.aiia.ed.ac.uk:/pub/packages/wxclips>

EXSYS¹¹⁴ este produsul unei companii americane (Exsys Inc.) și a fost lansat pe piață în 1989, iar EXSYS DEVELOPER este cea mai recentă versiune a sa. Este destinat cu precădere celor ce doresc să-și creeze aplicații proprii foarte rapid. Are în structura sa două componente mari:

- generatorul propriu-zis sau editorul de reguli de producție, pentru crearea și modificarea unei baze de cunoștințe

- componenta RUNTIME sau “de consultare” care permite utilizarea bazei de cunoștințe create anterior

EXSYS a fost elaborat pentru calculatoare IBM-PC, operează cu reguli IF-THEN-ELSE, are un motor de inferențe capabil să funcționeze după strategia de control înainte, înapoi și mixtă. Dispune de posibilități multiple de achiziție a cunoașterii, inclusiv pe căi probabilistice, permițând crearea unor sisteme expert cu mii de reguli. Admite și formule matematice și logice, complexe, realizate cu operatori logici și relaționali, care pot fi încorporate în reguli.

Editorul de reguli are posibilitatea de a modifica sau șterge reguli deja editate și este capabil de verificarea și validarea regulilor pe măsura introducerii lor în baza de cunoștințe. După terminarea achiziției regulilor, acestea trec la compilarea automată, pentru traducere în limbaj mașină, apoi li se face verificarea consistenței, iar dacă sunt în conflict cu alte reguli va fi solicitată reformularea lor. Dacă nici un conflict nu s-a constatat, utilizatorul proiectant dispune de patru opțiuni:

- inserarea sau editarea unei noi reguli
- salvarea tuturor regulilor pe disc sau alt suport
- imprimarea regulilor împreună cu o listă a referințelor încrucișate, dacă este cazul
- execuția setului de reguli

După lansarea motorului de inferențe, controlul său se realizează cu interfața utilizator, deosebit de prietenoasă și cu multiple posibilități de execuție sau de editare. Motorul de inferențe este dotat cu mai multe module, care se pot conecta la diferite echipamente de verificare automată a mediului înconjurător, senzori, software de calcul tabelar, software pentru grafică, SGBD relaționale, sintetizator vocal și orice programe externe. Toate aceste module intră în acțiune la momentul oportun, conform dorinței proiectantului sau utilizatorului. La rândul său, EXSYS poate intra în acțiune dacă este apelat de către un program extern sau o aplicație informatică inclusiv de tip clasic.

EXSYS cuprinde un proces de achiziție a cunoașterii în cinci pași:

¹¹⁴ [***] Exsys Developer <http://www.exsys.com/Exdev.html>

-determinarea de către cognitician a subiectului bazei de cunoștințe și identificarea unui expert în domeniu

- descoperirea cunoașterii de care dispune expertul, pe parcursul mai multor interviuri
- cunoașterea captată este formalizată în reguli de producție, care fac obiectul bazei de cunoștințe
- se construiește baza de cunoștințe, prin una sau mai multe sesiuni de editare a regulilor
- se testează baza de cunoștințe astfel obținută pentru a constata corectitudinea cunoașterii achiziționate

Regula de producție în EXSYS are 6 componente:

IF <premiză>

THEN <concluzie1>

ELSE <concluzie2>

NOTE <comentariu>

REFERENCE <comentariu>

NAME <nume>

Ultimele patru componente sunt opționale. Partea de IF se creează prin combinații de calificatori și valori asociate. Partea de THEN se creează prin combinarea unor opțiuni și a unor probabilități, considerate factori de certitudine. EXSYS oferă cinci metode pentru valorile factorilor de certitudine: Yes/No, [0,10], [-100,+100], Incr/Decr, Custom Formula și Fuzzy. Limitele la stânga ale fiecărui interval înseamnă incertitudine absolută, iar cele la dreapta înseamnă certitudine absolută.

EXSYS preferă mai întâi să infereze cunoașterea din alte reguli decât să o solicite de la utilizator. Acest tip de inferență este specific strategiei de control înapoi. Dacă utilizatorul adresează întrebarea “Why?”, în timpul sesiunii de consultare, se vor afișa, ca explicație, regulile utilizate în lanțul inferențial. Când se doresc mai multe detalii explicative se poate introduce semnul “?”. Dacă o regulă s-a afișat, există posibilitatea întrebării de unde cunoaște sistemul că enunțurile IF sunt adevărate, prin tastarea numărului liniei corespunzătoare unei condiții. Se poate cere o explicație referitoare la una dintre expresiile matematice utilizate și se obține valoarea fiecărei variabile implicate.

Cogniticianul, în procesul de elaborare a sistemului expert trebuie să parcurgă următorii pași:

- introducerea numelui viitoarei baze de cunoștințe
- stabilirea parametrilor de lucru cu sistemul (subiectul bazei de cunoștințe, numele autorului, metoda de lucru cu factori de certitudine, modalitatea de combinare a acestor factori de certitudine, activarea afișării regulilor pe timpul execuției sistemului, modul de execuție a bazei de cunoștințe etc)
- introducerea scopurilor specifice bazei de cunoștințe, a calificatorilor, cu valorile asociate, cunoscute din analiza problemei
- definirea scopurilor specifice bazei de cunoștințe
- introducerea variabilelor sau formulelor cunoscute din analiza problemei și a regulilor specifice reprezentării cunoașterii
- execuția, testarea și validarea sistemului expert.

Toate aceste răspunsuri sunt preluate din opțiuni de tip “butoane radio”, în rubrici corespunzătoare sau ferestre special concepute pentru fiecare în parte. Pentru dezvoltarea sistemului expert proiectantul trebuie să cunoască foarte bine o serie de proceduri de operare cum ar fi: procedura de creare a bazei de cunoștințe, de adăugare a variabilelor, de editare/modificare a unei reguli, de mutare și ștergere a unei reguli, de execuție (consultare) a sistemului expert creat, de regăsire a unei baze de cunoștințe, de salvare și ieșire din generatorul de sisteme expert.

În momentul în care sistemul expert ajunge la soluția (soluțiile) sau concluzia problemei, el afișează o listă a soluțiilor posibile. Se pot afișa chiar notițe sau valori ale unor variabile evaluate de către sistem. După afișarea soluțiilor, există posibilitatea schimbării unui sau mai multor răspunsuri ale utilizatorului pentru a vedea efectul asupra soluțiilor.

Mobal¹¹⁵ este un sistem de dezvoltare a unor modele operaționale folosind o anumită reprezentare a datelor. Acest sistem cuprinde un sistem de achiziție manuală a cunoștințelor și inspecție a lor, un motor de inferență, metode de învățare pentru achiziție automată de cunoștințe și un sistem de recunoaștere pe baza cunoștințelor. Folosind acest sistem de achiziție de cunoștințe, se poate dezvolta un

¹¹⁵ [***] Mobal <http://www-2.cs.cmu.edu/afs/cs/project/ai-repository/ai/areas/learning/systems/mobal/0.html>

model pentru un domeniu pe baza unor reguli și fapte date. Se pot vizualiza cunoștințele introduse în fereastră text sau windows, modifica sau îmbogăți baza de cunoștințe. Dacă există contradicții în baza de cunoștințe datorită unor reguli sau fapte incorecte, există o unealtă ce permite localizarea problemei și corectarea ei. Este disponibil gratuit pentru folosire în scopuri academice prin ftp la adresa :

<ftp://ftp.gmd.de/gmd/mlt/Mobal/>

Mike¹¹⁶ (Micro Interpreter for Knowledge Engineering) este dezvoltat de UK Open University. Folosește reguli de raționare înainte și înapoi și un limbaj de reprezentare bazat pe cadre și succesiuni definite de utilizator. Explicații de tip « cum » sunt pentru execuția regulilor precum și explicații de tip « de ce ». Este scris în Prolog și este disponibil prin ftp la adresa :

hcrl.open.ac.uk

WindExS¹¹⁷ (Windows Expert System) a fost dezvoltat ca un sistem expert sub Windows ce folosește reguli de raționare înapoi. Are o arhitectură modulară ce permite adăugarea de noi module pentru a putea spori facilitățile sistemului. Sistemul deține un procesor de reguli, motor de inferență, modul de gestiune a mesajelor și un modul pentru bazele de cunoștințe.

RT-Expert¹¹⁸ a fost gândit ca un sistem expert shareware care să permită programatorilor C să integreze reguli aferente unui sistem expert în aplicațiile C/C++. RT-Expert constă într-un compilator de reguli care compilează regulile în cod C împreună cu o bibliotecă ce se constituie într-un motor de execuție a regulilor. Versiunea DOS este compatibilă cu compilatoarele Borland Turbo C, Borland C++ și Microsoft C/C++. Ediția personală se poate folosi în scopuri educaționale, de cercetare, dar pentru scopuri comerciale este nevoie de licența pentru versiunea profesională putând astfel dezvolta aplicații în mediul DOS, Windows sau Unix. RT-Expert este disponibil prin ftp la adresa :

world.std.com:/vendors/rtis/rtexpert

Acquire¹¹⁹ este un mediu complet de dezvoltare pentru creare și întreținere de aplicații bazate pe cunoștințe. Oferă o metodologie pas cu pas pentru ingineria cunoașterii ceea ce permite experților să se implice direct în structurarea și codarea cunoștințelor. Implicarea directă a experților crește calitatea, completitudinea și acuratețea cunoștințelor achiziționate, scade costul întreținerii și crește controlul asupra întregii aplicații. Acquire oferă un model de achiziție bazat pe recunoașterea structurilor. Cunoștințele sunt reprezentate ca și obiecte, reguli de producție și tabele de decizie. De asemenea oferă tratarea incertitudinii, rapoarte cu numeroase facilități precum și un sistem de help în format hipertext.

Aion Development System¹²⁰ rulează pe diverse sisteme, incluzând DOS, OS/2, SunOS, Windows și VMS. Folosește o reprezentare a cunoștințelor orientată obiect, reguli de raționare înainte și înapoi, grafice precum și posibilități de apeluri din alte limbaje de programare.

Angoss Knowledge Seeker¹²¹ se constituie ca instrument data-mining ce poate fi utilizat pentru a obține reguli pe baza deducerilor ce rezultă din relațiile cauză-efect din cadrul unor baze de date.

Art Enterprise¹²² este ultimul din familia mediilor de dezvoltare bazate pe reguli. Cuprinde caracteristici ce nu sunt cuprinse în C++ sau Smaltalk, o largă colecție de clase și acces la baze de date (folosind SQL sau prin ODBC). Acest mediu cuprinde un motor de inferență cu raționare înainte, dar există și posibilitatea de implementare a raționării înapoi. Acest sistem poate fi încapsulat în alte aplicații.

¹¹⁶ [***] MIKE http://kmi.open.ac.uk/people/marc/mike_text.html

¹¹⁷ [***] WindExS <http://www-2.cs.cmu.edu/Groups/AI/html/faqs/ai/expert/part1/faq-doc-5.html>

¹¹⁸ [***] RT Expert <http://www.tele.ucl.ac.be/ELEC2920/1997/agentsP/complet.html>

¹¹⁹ [***] ACQUIRE <http://www.iit.nrc.ca/fuzzy/files.html>

¹²⁰ [***] Platinum Technology Inc. <http://www.platinum.com>

¹²¹ [***] Angoss Knowledge Seeker <http://www.angoss.com>

¹²² [***] ART ENTERPRISE <http://www.ai-cbr.org/tools/brightware.html>

Arity Expert Development Package¹²³ este un sistem ce integrează reprezentarea bazată pe reguli cu cea bazată pe cadre, permițând utilizarea factorilor de certitudine.

Comdale¹²⁴ reprezintă un sistem expert în timp real dezvoltat pentru monitorizarea și controlul proceselor industriale. Oferă recomandări, concluzii și acțiuni de control fără a întrerupe procesul în desfășurare. Poate lucra cu cunoștințe și date incerte, are o arhitectură deschisă și funcționează în timp real. Are o configurație orientată obiect, permite lucrul în rețea, tratarea situațiilor de alarm, tratarea unor evenimente planificate.

Easy Reasoner¹²⁵ se constituie ca un instrument CBR (Case Base Retrieval) ce regăsește cazuri similare privind un caz nou dat. Extinde QBE (Query By Example) prin QBS (Query By Similarity). Indexează informațiile din bazele de date folosind arbori de decizie; poate accesa diverse baze de date prin ODBC. Informațiile sunt filtrate pentru eliminarea zgomotelor simplificând astfel arborii de decizie. Sistemul cataloghează și regăsește cazuri clasificate după similitudine. De asemenea utilizează mai mulți arbori de decizie, uneori pentru același câmp; tratează lipsa datelor și răspunsurile tip „Nu știu”.

Finexpert este un produs foarte elaborat și utilizează tehnicile cele mai recente de analiză financiară dezvoltate de departamentul “Finanțe” al Centrului de Învățământ Superior Economic din Franța. A fost realizat mai ales prin concursul experților contabili din Franța. FINEXPERT are ca scop efectuarea, pentru o întreprindere dată, a analizei rentabilității, a strategiei financiare și a riscurilor legate de exploatare. Utilizează motorul de inferențe Personal Plus Consultant al firmei Texas Instrument, asociat la o bază de cunoștințe ce cuprinde peste 200 de reguli.

Symede a fost dezvoltat ca un sistem expert de diagnostic al întreprinderii. S-a vrut un sistem de diagnostic global bazat pe o triplă serie de variabile: economice, juridice și financiare. Funcționează cu ajutorul generatorului GURU. Diagnosticul presupune înregistrarea documentelor contabile de sinteză pe ultimele trei exerciții utilizând 150 de parametri în baza de fapte. Baza de reguli este compusă din peste 600 de reguli. Modelul economic comportă date relativ la piață, concurență, produse, clienți, organizare comercială, etc. Modelul juridic traduce solvabilitatea clienților, structura patrimoniului întreprinderii, securitatea contractuală, capacitatea de adaptare, studiul riscurilor externe etc. Modelul financiar descrie activitatea din vânzări, rentabilitatea, echilibrele financiare și trezoreria.

Sistemul expert **Aide** aparține Centralei Băncii Franței. Baza de reguli comportă un bloc cu dominantă financiară orientat spre structura analizei întreprinderilor și un bloc cu dominantă economică axat în jurul competitivității, creșterii economice și strategiei. AIDE editează la sfârșit un raport de sinteză precizând situația întreprinderii, potențialul său, rezultatele obținute și strategia de urmat. Deși acest sistem este foarte performant, el nu are flexibilitatea unui expert uman.

Sedre (Systeme Expert de Diagnostic sur la Rentabilite des Entreprises) este un sistem expert care furnizează, în aproximativ 15 minute, un diagnostic economic și financiar al unei întreprinderi, plecând de la documentele contabile din ultimii doi sau trei ani și de la informații complementare financiare, economice și strategice. Informațiile contabile sunt prelucrate pentru a obține documentele necesare: bilanț sintetic, solduri intermediare de gestiune, tablouri de finanțare, indici etc. Rezultatele sunt interpretate și judecate în funcție de informațiile din mediul economic, despre activitate etc. Diagnosticul elaborat de SEDRE este un diagnostic global și sintetic, iar pentru fiecare domeniu specific în parte apare un diagnostic aprofundat și explicativ. SEDRE aduce un ajutor eficient tuturor persoanelor care doresc să formuleze un diagnostic plecând de la documentele contabile și au acces la informațiile contabile și economice din întreprindere: experți contabili, inspecți bancari, etc.

¹²³ [***] Arity Expert Development Package <http://www.arity.com/www.pl/products/ap.htm>

¹²⁴ [***] Comdale <http://mining.ubc.ca/product/>

¹²⁵ [***] Easy Reasoner <http://www.haley.com/>

Alte sisteme expert consacrate sunt: Activation Framework, Babylon, CBR Express, Cogsys, ProcessVision, Crystal, Cxpert, Eclipse, Emerald Empower Procedural Advisor, Esteem, Icarus etc.

4.3.4.2. Utilizarea sistemelor expert în domeniul detectării fraudelor pe piața bursieră

După spectaculoasele fraude financiare de la Enron, WorldCom, Tyco, Xerox, Waste Management, Bearings PLC, Sunbeam și altele, atât investitorii cât și organismele de control și-au pus din ce în ce mai acut problema prevenirii acestor fraude. Sistemele expert pot constitui instrumente care să sprijine toți acești factori în depistarea acestor cazuri.

Investitorii se pot confrunta și cu fraude care să se manifeste pe piețele reglementate pe care se tranzacționează valori mobiliare. Supravegherea pieței bursiere suferă de toate problemele procesării informațiilor inerente în orice domeniu unde volumul, acuratețea, oportunitatea, densitatea și complexitatea datelor depășesc posibilitățile umane, fiind încărcate de nesiguranță.

Câteva piețe care s-au asigurat de deschiderea lor, de cinstea și echitabilitatea pentru toți investitorii, au încercat să abordeze problemele procesării informației prin utilizarea unor baze de cunoștințe diverse. Din ce în ce mai mult aceste baze de cunoștințe au devenit mai ample, mai mari pentru a evita fenomenele de schimb nejuste în ceea ce privește cursul sau inițierea tranzacțiilor și pentru a detecta activități ilegale cum ar fi spălarea banilor. În multe privințe sarcina procesării automate pentru a identifica astfel de tranzacții este reversul prognozelor financiare: în loc de a încerca să se prezică anomaliile viitoare, analiștii, supraveghetorii bursei de valori trebuie să se uite înapoi pentru a detecta șabloanele, anomaliile sau relațiile care ar putea fi generate de informații neaccesibile pieței în general.

Sistemul trebuie automat să pondereze două scopuri conflictuale: să dea analiștilor tot ce au nevoie pentru a reprezenta activitățile dubioase și să evite bombardarea lor cu alerte și avertismente inutile. În cele mai multe privințe, determinarea suspiciunilor unei tranzacții sau tranzacționarea în general este foarte greu de stabilit empiric și e greu să o măsurăm.

Cu toate că expertiza este foarte greu de modelat cu reguli convenționale, burse ca și National Association Securities Dealers Automatic Quotation (NASDAQ) și New York Stock Exchange (NYSE) au încercat cel puțin să clasifice câte ceva din neregulile informaționale.

Dificultatea inerentă de modelare a comportamentului dinamic cu reguli și ajustarea lor astfel încât să schimbe șabloanele de activitate a făcut ca Bursa de Valori din Toronto să nu fie dispusă să se bazeze pe reguli. Toronto Stock Exchange, o piață mai degrabă mică, avea nevoie de alte tehnici mai flexibile, dinamice și capabile să învețe. Această tehnică este CBR (Case Based Reasoning), bazată pe raționamente.

Scopurile mereu prezente, dar conflictuale de a prinde toate anomaliile fără a bombarda analiștii cu un număr excesiv de alerte definesc clar orizontul soluției.

Principala sarcină este de a alege variațiile neobișnuite de preț sau de volum al tranzacțiilor pentru a asigura securitatea tranzacțiilor zilnice. În trecut se semnală, de exemplu, orice modificare de curs sau volum mai mare de 2%. Rezultatele constau în mii de alerte, puține fiind relevante sau interesante. În timp, analiștii au învățat să rezolve anomaliile cu o rată mai mare de acuratețe. O mișcare semnificativă a unei acțiuni era neobișnuită dacă era mai mare de 45% în volum și 2% în preț. Analistul trebuie să înceapă să strângă fapte noi, evenimente care să explice modificările neobișnuite. Toate aceste operații se pot realiza cu editoarele codificate hard. Analizând aceste editoare din cadrul principal al programului de tranzacționare, ele se dovedesc a fi singurele alternative pe care le cer resursele și costurile implementării mecanismului de filtrare care în acest mediu au fost adesea interzise. De aceea Toronto Stock Exchange a optat pentru o stație de lucru care producea aceste filtre pe datele luate din programul principal. Se desemnează devierile din tranzacții care indică activități neobișnuite sau ilicite, dar cu un grad ridicat de abstractizare. Legăturile dintre reguli s-au dezvoltat pentru a exprima cunoștințele despre mișcările de acțiuni, pentru a filtra mai bine alertele. Acest gen de logică se poate aplica acțiunilor de către expert pentru a determina eficacitatea tranzacțiilor corecte, bazate pe măsuri statistice sau euristice. Într-o piață turbulentă această metodă poate inunda analiștii cu alerte inutile și irelevante. Dificultățile de creare, rafinare și modificare a regulilor duc la probleme de dezvoltare a acestei metode:

- timp și resurse mari sunt cheltuite în dezvoltarea regulilor
- importante componente ale domeniului sunt frecvent neacoperite de aceste reguli

- presupun codificare hard, judecăți, intuiție
- structurarea regulilor cere un consum mare de timp, putând duce la ambiguități și contradicții în domeniu

Cunoașterea șabloanelor este o altă sarcină de monitorizare a pieței. Multe sisteme așteaptă să observe tipare în stabilirea prețului unei acțiuni sau în volumul tranzacțiilor. Dar aceste măsuri statistice sunt limitate în a evalua date non-numerice. Ori astăzi piața bursieră este frecvent influențată de factori care nu pot fi reprezentați ușor în ecuații. Astfel soluția așteptată de Toronto Stock Exchange a fost un mecanism bazat pe logica fuzzy care assemblează un proces nestructurat pe care îl utilizează analiștii. Situațiile și limitele lor determină un motor inductiv de predicție. Această metodă nu reclamă precizii extreme și nu necesită date complete.

O aplicație utilizată pe piața din Toronto este Intelligent Market Monitor (IMM), scrisă în C, care rulează pe o mașină UNIX. Este interfațată direct cu tranzacționarea. Acest sistem lucrează în două faze distincte:

1. Filtarea fluxului de date care intră (sub formă de formule matematice)
2. Fiecare tranzacție care trece de filtru este trimisă spre un model de interpretare care mărește baza de date și trece rezultatele într-un algoritm de decizie pentru tranzacții de alertă.

Sistemul necesită și alte informații referitoare la date despre tranzacțiile recente ale unui tip de titlu de valoare, evoluția cursului acțiunilor industriale etc. Tranzacțiile neobișnuite care nu au explicații bazate pe informațiile disponibile sunt considerate a fi esențial de investigat.

Cererile de rezolvare cer sisteme de rutine de mare acuratețe și foarte rapide. Filtrul de alertă poate procesa pe puțin 20 de tranzacții pe secundă, iar modelul poate evalua 6 tranzacții pe secundă depinzând de numărul de interogări necesare asupra bazelor de date. Toronto Stock Exchange este un start important pe calea progresului, pentru realizarea unor aplicații complexe în acest domeniu.

Bursa din Londra a implementat un sistem inteligent hibrid pentru detectarea tranzacțiilor dubioase numit MonITARS (Monitoring Insider Trading and Regulatory Surveillance). Acest sistem combină tehnologia algoritmilor genetici, logica fuzzy și rețelele neuronale în scopul supravegherii activității de tranzacționare (peste 100.000 de tranzacții bursiere zilnice, în care sunt implicați sute de agenți de bursă și milioane de investitori).

Din 1990 Federația Comunității de Valori Mobiliare a luat inițiativa de a dezvolta o rețea informațională (Pan Europa) în care erau incluse 12 burse europene. Sistemul prevedea:

- actualizarea tranzacțiilor pentru orice membru
- cursul valurilor europene
- rezultatele unor proiecte de cercetare
- elemente privitoare la FOREX și cursul acțiunilor

Exemplificând cazul Elveției de pe piața europeană, aici computerele și rețelele de comunicații au transformat cele șapte regiuni de piață bursieră într-o mare rețea națională, informatizată.

4.3.4.3. MarketMind¹²⁶

Acesta reprezintă un produs al companiei Intelligent Market Analytics și se constituie într-un sistem expert integrat, care pe baza unui proces de analiză poate oferi sugestii de vânzare sau cumpărare. Pe lângă regulile predefinite, în cazul în care utilizatorul dorește să decidă pe baza unor reguli proprii, sistemul permite adăugări de reguli. Ieșirile sistemului se obțin sub forma unor ferestre de alarmă. Sistemul folosește, pentru reflectarea stării unei anumite valori mobiliare, un sistem de săgeți verticale de diferite mărimi, orientate în sus pentru a reflecta o creștere a prețului sau o îmbunătățire a situației unei valori mobiliare, aspect pozitiv și orientate în jos pentru situația inversă. Se remarcă naturalețea și simplitatea acestor notații, menite să faciliteze utilizarea sistemului de investitor.

Fiecare regulă va determina numărul, direcția și mărimea săgeților corespunzătoare gradului de îndeplinire a respectivei reguli. De asemenea, pentru fiecare regulă va rezulta o anumită condiție de alarmă. În vederea determinării condiției finale se va proceda la calcularea mediei ponderate cu

¹²⁶ [***] MarketMind <http://www.marketmind.com>

importanța regulilor. Rezultatele utilizării sistemului MarketMind sunt vizibile pe ecran prin intermediul ferestrei de alarmă, care poate conține un număr nelimitat de celule, unde fiecare celulă are un nume (simbolul valorii mobiliare) și un număr de săgeți (condiții de alarmă). Fereastra de alarmă este un ghid pentru investitor și ea poate fi setată să se actualizeze după cerințele utilizatorului (o dată/zi, o dată/oră etc).

Trebuie subliniat și faptul că MarketMind dispune de un sistem de grafice, tabele și explicații menite să satisfacă cerințele de informare ale investitorului și să-l conducă spre o decizie cât mai fericită în gestiunea portofoliului.

4.3.4.4. Investment Advisor System¹²⁷

Produsul este un sistem de consultanță în plasamente elaborat de firma Nomura Securities sub forma unui sistem expert. Această firmă de investiții japoneză se distinge prin popularizarea pe care o face produselor și serviciilor sale printr-o largă rețea de distribuție a informațiilor economice și financiare.

Sistemul permite analiza profilului clientului, adresând întrebări de genul: motivul pentru care clientul dorește să facă investiții, vârsta, câștigul lunar, existența unor obligații financiare, ce valori mobiliare deține, venitul acestora, maturitatea, venitul dorit și modalitatea de creștere (încet și constant, mai agresiv sau foarte agresiv). Întrebările sunt adresate interactiv, prin chestionare simple și scurte. Sistemul prezintă toate produsele financiare oferite de Nomura Securities.

În acest sens, utilizatorul poate selecta produsul pe care îl consideră cel mai adecvat scopului său investițional, după care unul din modulele sistemului expert oferă sugestii de investiții. În funcție de răspunsurile clientului, sistemul se va orienta spre un portofoliu agresiv sau din contra, spre unul conservativ.

4.3.4.5. Integrative Stock System

Elaborat de Yamaichi Securities, Integrative Stock System este un sistem de gestiune a investițiilor în acțiuni, capabil să efectueze analize de piață multidimensionale, fiind folosit de investitori pe piața de capital japoneză și americană. ISS stabilește prețul corect al acțiunilor luând în considerare microfactori, macrofactori și alți factori legați de piață. Sistemul încearcă o previzionare a comportamentului viitor al acțiunilor, pe baza unor modele și are 4 părți componente, prin care se realizează:

- evaluarea raportului risc/venit, bazată pe analiza multidimensională a microeconomiei și macroeconomiei
- stabilirea trendului diferiților factori ai pieței
- calcularea simultană a riscului și venitului
- construirea unui portofoliu care să satisfacă cerințele clienților.

Sistemul folosește trei parametri: primul indică dacă prețul de piață al unei acțiuni este corect stabilit; al doilea arată sensibilitatea acțiunii sau portofoliului la modificarea indexului pieței, iar ultimul indică sensibilitatea pieței față de factorii macroeconomici.

Construirea portofoliului cu ajutorul lui ISS se realizează în următoarele etape:

1. *crearea unei liste principale* –ISS selectează automat reprezentanți ai fiecărei industrii care sunt cotați în prima secțiune a bursei. Criteriul folosit este ca indicatorii financiari ai acestora (lichiditate, solvabilitate) să depășească nivelul mediu.
2. *construirea portofoliului* –având la bază specificațiile clientului privind mărimea riscului și rata venitului, din lista stabilită la prima etapă se vor stabili acțiunile ce vor intra în compoziția portofoliului, precum și volumul acestora.
3. *revizionarea portofoliului* –se realizează automat, semnalând dacă sunt necesare modificări

4.3.4.6. Integrative Expert System

¹²⁷ [***] Investment Adviser <http://www.iard.com/Default.asp>

Elaborat de Yamaichi Securities, Integrative Expert System oferă utilizatorului indicații de vânzare/cumpărare relativ la investițiile în valori mobiliare și cuprinde două părți:

1. *sistemul de alarmă* -are rolul de a da semnale de vânzare sau cumpărare atunci când există indicii că vor avea loc schimbări pe piață. Acest modul clasifică toate companiile cotate în prima secțiune a bursei din Tokyo în 65 de sectoare industriale și ține cont de sensibilitatea fiecărui sector în raport cu indicatorii macroeconomici. O firmă și acțiunile sale poate face parte din mai multe sectoare, cu o anumită pondere în fiecare, determinată de structura vânzărilor. Modulul lucrează on-line și are acces la o bază de date care listează ultimul curs al acțiunilor și statistici financiare.

2. *procesul de raționare* –permite aplicarea anumitor reguli (precum ar fi: vinde atunci când PER “Price to Earnings Ratio” este mare sau vinde dacă există abateri semnificative ale cursului de la o medie mobilă dată). Fiecare acțiune poate avea mai multe semnale de vânzare sau cumpărare gradate, în funcție de numărul de sectoare cărora le aparține. Gradul corespunzător unei acțiuni este media ponderată cu vânzările divizate pe sectoare. Rezultatul procesului de raționare se reprezintă grafic pentru fiecare acțiune. Fiecare grafic reprezintă doar acele grade și volume care vor declanșa alarma. Partea if-then din reguli ține cont de firma emitentă și de ramura industrială din care face parte. Regulile cuprind standarde care specifică ce semnale de vânzare sau cumpărare trebuie emise pentru fiecare acțiune dacă apar schimbări pe piață. Regulile utilizate pot fi de următoarele tipuri:

- *macroreguli* –vizează standarde referitoare la trendurile economice generale care afectează piața de capital, aspecte privind alegerea industriei și acțiunii. Folosește variabile cum ar fi prețul mărfurilor, evoluția ratei dobânzii etc.
- *reguli privind ramura industrială* -reflectă evoluția unor indicatori care caracterizează ramurile industriale
- *microreguli* –permit dezvoltarea standardelor necesare pentru selectarea acțiunilor individuale, indiferent de datele referitoare la industrie. Variabilele folosite sunt: abaterea cursului acțiunii față de media mobilă, riscul nediversificabil.
- *reguli individuale* –arată preferințele individuale și restricțiile impuse de investitori și autorități.

Se calculează o medie ponderată a semnalelor de cumpărare sau vânzare, ponderea reflectând viziunea investitorului asupra pieței. De exemplu, dacă investitorul consideră că piața este orientată spre câștiguri mai ales din diferențe de preț, microregulile vor deține o pondere mai mare.

IES folosește un subsistem pentru gestiunea regulilor care analizează structura regulilor, sensibilitatea lor și poate adăuga reguli noi. Sensibilitatea regulilor se determină prin studierea efectelor pe care le are modificarea unei reguli asupra semnalelor de vânzare sau cumpărare. Sistemul poate fi folosit nu numai pentru evaluarea noilor investiții, dar și pentru verificarea portofoliilor existente, în vederea maximizării ratei venitului proprie acestuia. Ca rezultat se poate obține o privire de ansamblu asupra portofoliului prin evaluarea fiecărei acțiuni componente.

4.3.4.7. Signal¹²⁸

Acest sistem al companiei Data Broadcasting Corporation servește investitorilor asistându-i în gestiunea portofoliului propriu, oferindu-le cotații, știri și analize de piață și oferă informații 24 de ore pe zi. El afișează cotațiile unui număr nelimitat de titluri și avertizează prin semnale sonore când cursul atinge anumite limite oferind astfel facilități de acțiune rapidă.

Se remarcă în același timp faptul că produsul este ușor de configurat potrivit opțiunilor fiecărui utilizator în parte. Caracteristica de bază a sistemului constă în faptul că oferă acces spre surse de informații precum Dow Jones, IFN, Futures World News (FWN) și altele, evoluții a peste 250 de indici, rapoarte zilnice ce conțin sfaturi pentru investitori, știri și noutăți despre piață, comentarii.

Deoarece poate prelucra un volum mare de date, permite efectuarea unor analize rapide. Este un bun manager de portofoliu, în orice moment utilizatorul putând afla cât valorează investițiile sale.

¹²⁸ [***] Esignal <http://www.esignal.co>

4.3.4.8. ITA -Intelligent Trading Assistants

Reprezintă un sistem menit să simplifice activitatea traderilor, prin executarea automată a unor operații de rutină, efectuarea calculelor complexe necesare, atenționarea traderului asupra pericolelor ce apar odată cu modificarea condițiilor de pe piață. Oferă propuneri privind:

1. numărul și tipul valorilor mobiliare care trebuie vândute și cumpărate
2. care sunt limitele de preț între care trebuie să execute tranzacțiile etc.

ITA cuprinde module pentru monitorizarea prețului și volumului mișcărilor pentru toate acțiunile, obligațiunile și alte titluri de valoare pentru care investitorul prezintă interes și module ce ilustrează părerea expertului relativ la rezultatele obținute de modulul anterior. De asemenea, sistemul permite identificarea celei mai bune cerere și celei mai bune oferte pentru o anumită valoare mobilă, compararea piețelor folosind grafice, astfel încât traderul să fie scutit de consultarea statisticilor. În același timp, rapoartele simple, bine documentate, cuprinzătoare și interactive indică traderului ce să vândă și ce să cumpere, asigurând întreținerea portofoliului, evaluarea riscului și venitului potențial și consultanță.

4.3.4.9. ZETA –Real time securities trading assistant

Sistemul asigură asistență în efectuarea tranzacțiilor prin determinarea acțiunilor care se mișcă în sens contrar grupei căreia aparține la bursa la care este cotate. Gruparea are la bază ramura industrială din care face parte firma. Această mișcare trebuie să se refere la un volum mare de acțiuni pentru a fi semnificativ din punct de vedere al trendului. ZETA aplică un set de reguli bazate pe cunoștințe asupra unui flux de date legate de piață obținute în timp real (ex: cursul, volumul tranzacțiilor, indicatori financiari). Dacă volumul tranzacționat dintr-o valoare mobilă este mai mare decât media, se analizează modificarea prețului în perioada imediat precedentă. Dacă se detectează o modificare considerabilă în volumul tranzacționat dintr-o acțiune, ZETA va examina modificarea prețului în minutele precedente. Dacă prețul a crescut, sistemul va sugera o cumpărare.

În cazul în care analiza va indica existența unor oportunități de investiție în anumite valori mobiliare, în prealabil se verifică media prețurilor precedente ale titlului în cauză, în comparație cu prețurile existente în sectorul de activitate din care face parte firma emitentă. Dacă prețul se modifică în sens contrar sectorului, se verifică și modificarea față de prețul de deschidere. Dacă mișcarea este contrară, sistemul propune din nou luarea poziției recomandate, dar menționează faptul că propunerea se bazează doar pe o modificare a cursului pe termen scurt. Sistemul monitorizează mișcările care au loc în vederea detectării imediate a unei modificări radicale a situației, ceea ce propune modificarea propunerii anterioare.

4.3.4.10. MIRI –Mitsubishi Research Institute¹²⁹

Cercetătorii de la Mitsubshi Research Institute au dezvoltat un sistem expert pentru tranzacționare ce permite aproximarea trendului pieței pentru ziua următoare. Sistemul se bazează pe construirea unui model, care se compară cu realitatea. Astfel se examinează acțiunile și obligațiunile, pentru a detecta în ce măsură se încadrează în model. Mitsubishi a dezvoltat și un alt sistem expert, destinat clasificării obligațiunilor și identificării celor mai eficiente dintre acestea prin care sistemul identifică principalele caracteristici ale fiecărei obligațiuni în parte, subliniază punctele tari și slabe, după care le compară.

4.3.4.11. Fund Manager¹³⁰

Este o aplicație complexă de managementul portofoliului investitorilor individuali. A fost proiectată pentru a ajuta investitorii în urmărirea și analiza acțiunilor, obligațiunilor, titlurilor fondurilor

¹²⁹ [***] Mitsubishi Research Institute <http://www.mri.co.jp/e/index.html>

¹³⁰ [***] Fund Manager <http://www.beiley.com/fundman/>

mutuale și a altor titluri de valoare cu o varietate de grafice și rapoarte. În 1997 a fost câștigătoarea premiului Ziff-Davis la categoria Finanțe. Este un produs shareware ce poate fi folosit sub orice platformă Windows.

Programul oferă facilități deosebite prin care se pot suprapune pe același grafic mai multe titluri simultan. Utilizatorul are posibilitatea să-și ajusteze culorile, fonturile după preferințele proprii. De asemenea aceeași facilitate o prezintă și rapoartele, utilizatorul având la dispoziție 100 de câmpuri disponibile pentru a fi incluse în rapoarte. O facilitate deosebită este aceea de a urmări evoluția pe același grafic a cumpărărilor, vânzărilor și a valorii investite.

4.3.5. Concluzii

Din problematica studiată despre sistemele expert, *autorul desprinde următoarele concluzii:*

Sistemele expert au adus un număr semnificativ de avantaje, printre care: creșterea cu un ordin de mărime a vitezei și complexității problemelor rezolvate, creșterea calității soluțiilor, scăderea numărului de erori, reducerea costurilor de producere a programelor, reducerea timpului de instruire a utilizatorilor, facilitarea activității de întreținere a programelor. De exemplu, sistemul expert XSEL de asistare a vânzărilor de la Digital Equipment Corporation a redus timpul de 3 ore necesar generării și configurării sistemelor cerute de clienți la 15 minute și numărul de sisteme incorecte generate în acest proces de la 30% la 1%. Conform DEC, sistemul XSEL a adus o economie de 70 milioane de dolari anual. Sistemul bazat pe cunoștințe Optex care proiectează lentilele aparatelor fotografice ale firmei Canon a crescut productivitatea activității de 12 ori. Compania de cărți de credit American Express a raportat că sistemul expert utilizat pentru validarea creditului acordat clienților săi a redus cu o treime numărul deciziilor eronate de neacordare a creditului, reducere estimată de companie ca reprezentând o economie de 27 de milioane de dolari anual. Sistemul *Performance Analyst* de la Departamentul de Sănătate și Servicii Sociale din Marea Britanie a redus timpul necesar evaluării unei cereri de la 2 ore la 9 minute, deci o creștere a productivității de 80 de ori.

Evaluarea creșterii productivității aduse de introducerea unei noi tehnologii trebuie întotdeauna făcută în raport cu costul achiziției și introducerii tehnologiei. Costul dezvoltării unui sistem expert poate fi detaliat în raport cu etapele realizării sistemului, astfel: identificarea și caracterizarea problemei, planificarea proiectului, achiziția cunoștințelor, proiectarea, achiziția tehnologiei adecvate programului, dezvoltarea sistemului, integrarea sistemului în sistemul logistic existent, verificarea, validarea și testarea prototipului, întreținerea bazei de cunoștințe. Proiectarea sistemului și achiziția tehnologiei sunt activitățile asupra cărora se concentrează în principal proiectanții, dar cele trei activități care par a domina costul introducerii unui astfel de sistem sunt achiziția cunoștințelor, integrarea sistemului și întreținerea bazei de cunoștințe. Costul achiziției cunoștințelor crește dramatic dacă dezvoltarea sistemului necesită prezența îndelungată a unui expert sau grup de experți de excepție. Costul integrării sistemului crește neliniar cu numărul și complexitatea modulelor ce trebuie integrate, acest cost depășind uneori costul dezvoltării componentei bazate pe cunoștințe. În sfârșit, costul întreținerii bazei de cunoștințe poate domina costul ciclului de viață al întregului sistem, în special în cazul în care cunoștințele implicate au o dinamică ridicată.

Pornind de la ideea că analiza activității bursiere se realizează numai prin sisteme bazate pe incertitudine, adică probabilități sau sisteme fuzzy și având în vedere că în România nu există statistici pe baza cărora să se poată realiza sisteme stohastice, considerăm că cele mai potrivite sisteme sunt cele bazate pe teoria fuzzy. De altfel, în acest domeniu există o serie de reguli euristice, cum ar fi cele bazate pe unele studii, care au arătat că întotdeauna este preferabil un portofoliu administrat unuia format din titluri selecționate aleator.

Aplicațiile Inteligenței Artificiale în domeniul financiar cunosc o mare răspândire și duc prin utilizarea lor la rezultate spectaculoase. În domeniul bursier, volumul mare de informații care se schimbă rapid necesită utilizarea Inteligenței Artificiale pentru prelucrarea lor și fructificarea rezultatelor. Având în vedere că datele oferite de piața bursieră și piața RASDAQ sunt în continuă creștere, este previzibil ca în scurt timp volumul lor să nu mai poată fi gestionat prin baze de date clasice, necesitând depozite sau antrepozite de date. Într-un astfel de sistem, apreciem necesitatea asocierii bazei de cunoștințe a unui

sistem Knowledge Discovery in Databases (KDD)¹³¹. Această tehnologie presupune sintetizarea unor tehnici variate aparținând statisticii, matematicii, logicii, informaticii folosite în scopul extragerii de informații utile din cantități mari de date corupte de zgomote informaționale, informații superflue sau incomplete. Interogările inteligente (exemplu KDD) în marile baze de date (data warehouse) oferite de aplicațiile inteligente lasă în urmă vechile tehnologii depășite din punct de vedere al eficienței, securității, dar mai ales din punct de vedere al tratării datelor (date incomplete, eronate, valori nule etc).

¹³¹ [Nițchi97] Nițchi S.I., Nițchi R. -Data mining, o nouă eră în informatică, BYTE 3,1997, 117-121.

CAPITOLUL 5. CONCEPEREA, PROIECTAREA ȘI IMPLEMENTAREA ELABORAREA UNUI SISTEM DE ASISTARE A DECIZIILOR ÎN DOMENIUL ANALIZEI BURSIERE

5.1. “STOCKASSIST” - SISTEM DE ASISTARE A DECIZIILOR ÎN DOMENIUL ANALIZEI BURSIERE

Această aplicație își propune să amelioreze structura informațiilor prezentate utilizatorilor de către Bursa de Valori, care nu efectuează analize asupra datelor brute rezultate în urma procesului de tranzacționare, ci doar le prezintă într-o formă structurată și centralizată. Obiectivele acestui sistem informatic sunt pe de o parte preluarea și punerea la dispoziția utilizatorilor a datelor bursiere, iar pe de altă parte efectuarea unor analize asupra datelor respective pentru a furniza un suport de decizie eventualilor investitori.

Datele de intrare preluate în sistem provin de pe site-urile Bursei de Valori și Rasdaq, care au obligația prin lege să facă publice datele referitoare la tranzacțiile ce au avut loc. Dezavantajul este că datele nu sunt preluate on-line ceea ce poate duce la scăderea eficienței aplicației care în mod normal ar trebui să prelucrez date în timp real. Acest lucru uneori nu e întotdeauna posibil nici în țări cu piețe bursiere dezvoltate și informatizate.

Potențialii utilizatori ai acestei aplicații sunt traderii, agenții de valori mobiliare și consultanții în valori mobiliare, cărora le oferă informații în vederea susținerii deciziilor de investiții pe care aceștia trebuie să le ia. Această aplicație nu va înlocui total analistul ci doar va furniza informații pe baza cărora utilizatorii să poată lua o decizie oportună, rațională și bine fundamentată teoretic.

Din punct de vedere al implementării, aplicația “StockAssist” a fost elaborată în Visual FOXPRO 8.0, având o interfață prietenoasă cu utilizatorul și dispunând de un sistem de meniuri adecvat.

Principalele caracteristici ale mediului de programare “Visual Foxpro” sunt următoarele¹³²:

- Se bazează pe *modelul programării orientate spre obiecte* (POO Programming Oriented Object) și a programării conduse de evenimente. Ca urmare, elementul de bază cu care lucrează este obiectul, care poate fi o fereastră, un buton de comandă, o relație între tabele etc. În acest context, un program în Visual Foxpro include o serie de definiții de date (proprietăți) și de algoritmi de prelucrare a acestora (metode, proceduri). Metodele asociate unui obiect sunt executate fie printr-o comandă explicită fie la apariția unui anumit eveniment (click cu mouse-ul, apăsarea unei taste, mutarea sau redimensionarea unei ferestre etc).

- Ca tehnici sau instrumente de programare, mediul Visual Foxpro folosește *programarea vizuală și tehnici de programare vizuale*. Acesta permite dezvoltatorilor folosirea unor obiecte vizuale sau asistenți de tipul Form Wizard, Graph Wizard, Report Wizard, care realizează generarea automată a programelor corespunzătoare.

- Este un *mediu integrat de programare sub Windows*, ce include un gestionar al aplicațiilor create în Visual Foxpro, numit Project Manager care grupează toate componentele unei aplicații (date, forme, rapoarte, clase, cod și obiecte FoxPro) și pune la dispoziția dezvoltatorilor unelte necesare efectuării acestor operații, cum ar fi generatoare (Designers), constructori (Builders) și asistenți speciali (Wizards), fiecare având un anumit domeniu de utilizare.

- Este un *sistem relațional de gestiune a bazelor de date*, asigurând crearea de relații statice (permanente) între tabelele din baza de date.

Visual Foxpro se constituie într-un instrument puternic de dezvoltare a aplicațiilor, ce asigură compatibilitate de jos în sus, deci toate aplicațiile dezvoltate în xBase sau Foxpro din versiunile anterioare pot fi rulate și în cele superioare. Aceasta reprezintă o facilitate importantă, având în vedere că multe firme și instituții din România au avut sau încă mai utilizează aplicații Foxpro ce necesită portarea lor la o versiune mai evoluată.

Partea de grafică a programului a fost concepută în Microsoft EXCEL 2000 pornind de la considerentul că în afara funcției uzuale de tratare a datelor în formă tabelară (așa numita foaie electronică

¹³² [Ghișoiu04] Ghișoiu N., Nițchi S, Nițchi A.R., Tomai N. –Baze de date și programare, Editura Risoprint, 2004.

de calcul -*spreadsheet*), acesta are o deosebită maleabilitate în prezentarea grafică a informației. De aceea, pentru partea grafică a programului am ales EXCEL.

Produsul se prezintă sub forma unui kit de instalare (INSTAL.BAT) care instalează programul pe hard-diskul calculatorului. La prima lansare a programului, are loc înregistrarea primului utilizator al programului. Acesta introduce un cod și o parolă. Ulterior, se pot înregistra și alți utilizatori, fiecare cu propria sa parolă. Din motive de securitate a aplicației, programul solicită la lansare codul și parola, în caz de nereușită, nepermițându-se accesul în program. De asemenea programul include un algoritm ce realizează modificarea parolei, astfel că vizualizarea tabeli în care se găsesc parolele nu permite cunoașterea acestora.

Datorită faptului că programul utilizează în cadrul procedurilor sale data curentă, la lansarea programului are loc verificarea datei interne a calculatorului; în cazul în care aceasta nu este cea reală, programul permite modificarea acesteia.

5.2. SITUAȚIILE FURNIZATE

Situațiile de ieșire furnizate de aplicație “StockAssist” constituie suportul de asistare a deciziei de investiție pe care analistul trebuie să o efectueze. Aceste situații de ieșire au la bază suportul teoretic prezentat în capitolul trei, bazându-se pe indicatorii analizei fundamentale, pe analiza tehnică și pe analiza bazată pe teoria piețelor eficiente.

Sistemul furnizează următoarele situații de ieșire:

Anexa 2	Indicatori statistici
Anexa 3	Analiza statistică
Anexa 4	Recomandări de investiții
Anexa 5	Simulare Random-Walk; Legea normală redusă de probabilitate
Anexa 6	Analiza grafică a cursului unei acțiuni
Anexa 7	Analiza cu ajutorul mediei mobile
Anexa 8	Valoarea actualizată a obligațiunilor
Anexa 9	Volatilitatea obligațiunilor
Anexa 10	Durata obligațiunilor
Anexa 11	Recomandarea unui portofoliu de acțiuni
Anexa 12	Volatilitatea unei acțiuni
Anexa 13	Coeficientul de corelație
Anexa 14	Evoluția cursului pe o perioadă dată

Fiecare din aceste situații se obține activând o opțiune din unul din submeniurile aplicației. În anexele corespunzătoare se găsesc exemplificări ale acestor situații, cu rezultatele ce se obțin.

5.3. STRUCTURA FIȘIERELOR DIN BAZA DE DATE

Baza de date proiectată (intitulată Analiza) conține următoarele tabele (figura 42):

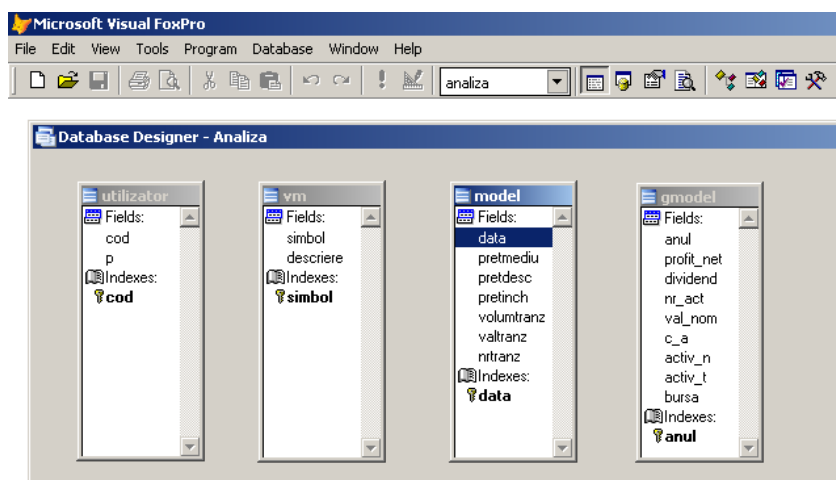


Fig.42 Structura bazei de date a aplicației “StockAssist”

Tabelele din această bază de date au următoarea structură:

- 1) **UTILIZATOR.DBF**
- | | | |
|-----|---------------|--------------------|
| Cod | Character(3) | -cheie de indexare |
| P | Character(10) | |

Câmpul Cod conține codul asociat fiecărui utilizator, iar P conține parola

- 2) **VM.DBF**
- | | | |
|-----------|---------------|--------------------|
| Simbol | Character(4) | -cheie de indexare |
| Descriere | Character(30) | |

Câmpul Simbol conține simbolul atașat fiecărei valori mobiliare, iar Descriere conține numele complet al societății emitente.

- 3) **MODEL.DBF**
- | | | |
|------------|--------------|--------------------|
| Data | Date | -cheie de indexare |
| Pretmediu | Numeric(9,2) | |
| Pretdesc | Numeric(9,2) | |
| Pretinch | Numeric(9,2) | |
| Volumtranz | Numeric(9) | |
| Valtranz | Numeric(13) | |
| Nrtranz | Numeric(7) | |

Pe baza acestui model, la adăugarea fiecărei valori mobiliare la listă se crează un fișier având ca nume simbolul asociat și un fișier index cu același nume.

- 4) **GMODEL.DBF**
- | | | |
|------------|-------------|--------------------|
| Anul | Numeric(4) | -cheie de indexare |
| Profit_net | Numeric(15) | |
| Dividend | Numeric(15) | |
| Nr_act | Numeric(12) | |
| Val_nom | Numeric(6) | |
| C_a | Numeric(15) | |
| Activ_n | Numeric(15) | |
| Activ_t | Numeric(15) | |
| Bursa | Logical | |

Pe baza acestui model, la adăugarea fiecărei valori mobiliare la listă se crează un fișier având ca nume simbolul asociat și un fișier index cu același nume.

5.4. INTERFAȚA SISTEMULUI

Prin interfață înțelegem orice comunicare între utilizator și computer, fie ea directă (în sensul că interacțiunea este controlată pe întreg parcursul ei, de pildă prin ferestre de dialog), sau indirectă (în sensul că invocă procese care decurg în fundal). Interacțiunea poate fi privită ca un dialog între utilizator și computer. Evident că în cazul acestui dialog, alegerea tipului de interactivitate are un efect puternic asupra naturii acestui dialog. Cele mai răspândite tipuri de interactivitate prezente la sistemele de calcul sunt realizate prin următoarele mijloace:

- linie comandă
- meniuri
- limbaj natural
- dialog (ferestre de dialog și interogări)
- forme și foi de calcul

Interfața utilizator este unul dintre elementele de bază ale produselor soft și, de regulă, un criteriu decisiv în evaluarea unui asemenea produs. O interfață dificil de utilizat afectează serios funcționalitatea sistemului. Astfel:

1. Interfața trebuie să utilizeze termeni și concepte familiare utilizatorilor vizati.
2. Interfața trebuie să fie consecventă .
3. Utilizatorul nu trebuie să fie "surprins" de sistem.
4. Interfața trebuie să includă un mecanism care să permită utilizatorilor corectarea erorilor.
5. Interfața trebuie să încorporeze o formă sau alta de ghidare a utilizatorului.

Primul principiu este mai curând o regulă de bun simț. De cele mai multe ori, bunul simț este un excelent sfătuitor. Un utilizator nu trebuie să se adapteze la interfața pe care o construiești doar pentru că aceasta este mai ușor de implementat.

Consecvența interfeței enunțată în al doilea principiu se referă la sistemul de comenzi și meniuri care în întregul produs soft trebuie să aibă același format, aparență și mod de funcționare. Acest lucru conduce la reducerea timpului de învățare.

Cel de al treilea principiu rezultă dintr-un proces natural care se produce cu utilizatorul. Pe măsură ce acesta întrebuințează un sistem, își construiește o reprezentare mintală a sistemului și în continuare se așteaptă ca sistemul să se încadreze în această reprezentare.

Cel de al patrulea principiu este ușor de înțeles: "Erare humanum est...". Designul interfeței trebuie să minimizeze apariția acestor greșeli. O modalitate este, de pildă, utilizarea meniurilor prin care utilizatorul este scutit de corvoada (supusă greșelilor) de a tasta comenzile. Există două căi principale pentru a face utilizatorului viața mai ușoară:

- Cererea unei confirmări pentru acțiunile distructive
- Includerea unei facilități de revenire asupra unei comenzi date (undo)

Ultimul principiu se referă la mâna de ajutor (help) care trebuie acordată utilizatorului. Modalitățile de a face acest lucru sunt diferite, de la "tururi cu ghid" prin sistem până la descrierea textuală a facilităților sistemului, structurată pe diverse criterii.

Aceste principii sugerează că designul unei interfețe trebuie să fie centrat pe utilizator. La urma urmelor, computerele nu sunt decât niște unelte cu care utilizatorii încearcă să-și rezolve o parte din probleme. Avantajul unei interfețe creată după modelul sarcinilor pe care le are de îndeplinit utilizatorul constă în învățarea rapidă și eficientă.

În cazul aplicației de față, interfața sistemului se compune dintr-un meniu combinat cu 6 opțiuni bară, fiecare opțiune având atașată câte un submeniu vertical așa cum se vede și în figura 43:

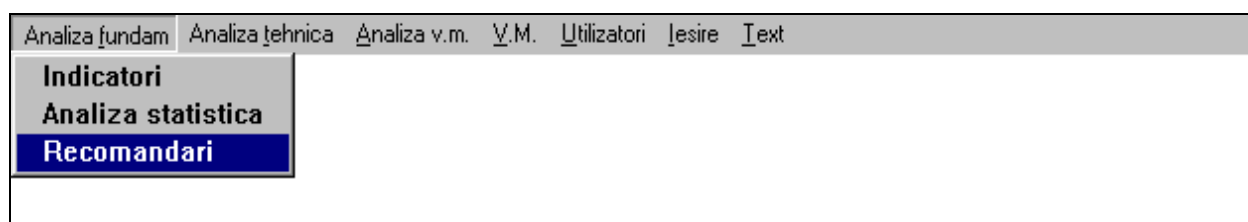


Fig. 43 Structura meniului aplicației "StockAssist"

Structura meniului bară și a submeniurilor este următoarea:

- Analiza fundamentală cuprinde:
 - Indicatori
 - Analiza statistică
 - Recomandări
- Analiza tehnică cuprinde :
 - Analiza grafică
 - Analiza bazată pe EMH (random-walk)
- Analiza valorilor mobiliare cuprinde:
 - Volatilitatea acțiunilor
 - Riscul acțiunilor
 - Valoarea actualizată a obligațiunilor
 - Volatilitatea obligațiunilor
 - Durata obligațiunilor
 - Recomandări
- Valori mobiliare cuprinde:
 - Adăugare valori mobiliare la listă
 - Ștergere valori mobiliare din listă
 - Introducere cursuri valori mobiliare
 - Modificare cursuri valori mobiliare
 - Introducere date (anual)
 - Modificare date
 - Import date
 - Situații
- Utilizatori cuprinde:
 - Adăugare utilizatori
 - Ștergere utilizatori
 - Lista utilizatorilor
 - Modificare parolă utilizatori
- ieșire

Această structură este prezentată în anexa 1.

5.5. PROIECTAREA MODULELOR

Aplicația “StockAssist” cuprinde următoarele module:

Modulul Indicatori –permite obținerea unei situații în care sunt prezentați indicatorii urmăriți și valorile corespunzătoare pe baza selecției valorii mobiliare dorite și a cursului introdus.

Modulul Analiza statistică –scopul lui este de a obține principalii indicatori ai analizei statistice pe baza selecției valorii mobiliare dorite, a începutului și sfârșitului perioadei de calcul.

Modulul Recomandări –stabilește o configurație optimă a portofoliului de valori mobiliare pe baza coeficienților de importanță acordați de utilizator aferenți celor 8 indicatori urmăriți. De asemenea este prezentată evoluția fiecărui indicator în timp pentru valorile mobiliare urmărite, sistemul recomandând în fiecare caz care este cea mai bună investiție din punctul de vedere al respectivului indicator. Toate aceste recomandări au la bază o perioadă de timp stabilită de utilizator (se introduc numărul de ani luat în calcul precum și anul cu care se încheie perioada).

Modulul Analiza grafică –permite analiza grafică a valorii mobiliare selectate. Lansează EXCEL în execuție și deschide fișierul angraf.xls. Acesta conține două foi de calcul:

- Analiza grafică -cu ajutorul acesteia se studiază evoluția în timp a cursului unei acțiuni.
- Medie mobilă -urmărește graficul mediei mobile pentru o anumită valoare mobilă.

Modulul Analiza bazată pe EMH –realizează analiza bazată pe teoria piețelor eficiente. Lansează EXCEL în execuție și deschide fișierul anemh.xls. Acesta conține la rândul lui două foi de calcul:

- Random-Walk -permite ca, pe baza unei valori de pornire să simuleze o evoluție aleatoare a cursului unei acțiuni în timp.
- Legea normală -prezintă distribuția legii normale reduse. Din studiile efectuate asupra rentabilității s-a ajuns la concluzia că legea normală constituie o bună aproximație a distribuției observate.

Modulul Volatilitatea acțiunilor -calculează rentabilitatea individuală și rentabilitatea pieței de-a lungul perioadei urmărite, coeficienții alfa și beta, ecuația de regresie și semnificația rezultatelor. Se introduc: numărul de ani pentru care se urmărește volatilitatea, anul cu care se încheie perioada și se selectează valoarea mobilă dorită.

Modulul Riscul acțiunilor – permite calculul coeficientului de corelație și explicația semnificației acestuia. Se introduc: numărul de ani pentru care se urmărește acest indicator, anul cu care se încheie perioada, se selectează valoarea mobilă dorită.

Modulul Valoarea actualizată a obligațiunilor –calculează valoarea actualizată a unei obligațiuni pe baza numărului de perioade luate în calcul, a valorii și a factorului de actualizare (pentru fiecare perioadă în parte).

Modulul Volatilitatea obligațiunilor –asigură calculul volatilității unei obligațiuni pe baza valorii nominale, a numărului de perioade luate în calcul, a ratei scadente din capital, dobânzii încasate și a factorului de actualizare (pentru fiecare perioadă în parte).

Modulul Durata obligațiunilor –calculează durata (duration) a unei obligațiuni pe baza acelorași date de mai sus

Modulul Recomandări –permite alcătuirea unui portofoliu optim pe baza ponderilor acordate de utilizator pentru fiecare categorie enumerată. De asemenea, se prezintă evoluția rentabilității de piață, a rentabilităților individuale, a coeficientului alfa și beta precum și a coeficientului de corelație de-a lungul perioadei urmărite.

Modulul Adăugare valori mobiliare la listă –realizează adăugarea unei noi valori mobiliare în listă. Se introduc simbolul (ce este unic, sistemul nu permite introducerea dublurilor) și descrierea, după care se salvează sau nu. În cazul în care se salvează, sistemul crează baza de date în care se vor reține cursurile zilnice și baza de date ce va conține datele comunicate anual de Adunarea Generală a Acționarilor (AGA), precum și indecșii asociați acestor baze de date. Datele se vor introduce prin opțiunile Introducere cursuri valori mobiliare respectiv Introducere date (anual).

Modulul Ștergere valori mobiliare din listă –realizează ștergerea din listă a unei valori mobiliare. Se selectează simbolul, după care se salvează sau nu ștergerea. În cazul ștergerii, vor fi eliminate și bazele de date corespunzătoare respectivei valori mobiliare.

Modulul Introducere cursuri valori mobiliare -permite introducerea datelor furnizate de rapoartele de tranzacționare oferite de Bursa de Valori sau de Rasdaq. Se selectează valoarea mobilă dorită, se introduc data, prețul mediu, prețul de deschidere și de închidere, volumul, valoarea și numărul tranzacțiilor, după care se salvează sau nu.

Modulul Modificare cursuri valori mobiliare -permite modificarea unor valori mai sus enumerate. Se selectează valoarea mobilă și se introduce data.

Modulul Introducere date (anual) -realizează introducerea datelor furnizate anual de AGA. Se selectează valoarea mobilă dorită, se introduce anul, profitul net, suma dividendelor, numărul acțiunilor emise, valoarea nominală, cifra de afaceri, activul net, activul total, dacă se tranzacționează în bursă sau pe piața extrabursieră, urmate de salvare sau nu.

Modulul Modificare date –permite modificarea valorilor de mai sus pe baza selecției valorii mobiliare și a anului de referință.

Modulul Import date -realizează importul datelor dintr-un fișier extern constituind raportul de tranzacționare oferit de pe site-ul Bursei de Valori.

Modulul Situații -afișează o situație ce cuprinde evoluția prețului respectivei valori mobiliare de-a lungul întregii perioade. Se selectează valoarea mobilă, se introduce începutul și sfârșitul perioadei urmărite.

Modulul Adăugare utilizatori –permite introducerea unui nou utilizator ce va avea acces la program. Această adăugare are loc prin introducerea unui cod (ce este unic pentru fiecare utilizator, sistemul nu permite coduri duble), a parolei și reconfirmarea ei urmate de salvare sau, după caz, la renunțare.

Modulul Ștergere utilizatori -realizează ștergerea unui anumit utilizator. Se face prin introducerea codului și a parolei respectivului utilizator, urmate de confirmarea ștergerii sau renunțare.

Modulul Lista utilizatorilor –permite afișarea listei utilizatorilor recunoscuți de sistem.

Modulul Modificare parolă utilizatori –realizează modificarea parolei unui utilizator. Se introduce codul, parola, noua parolă urmată de reconfirmarea ei.

Ieșire -permite părăsirea programului și revenirea la sistemul de operare.

Aceste module sunt prezentate în anexa 15.

5.6. APRECIERI PRIVIND EFICIENȚA ȘI FIABILITATEA SISTEMULUI INFORMATIC

Dacă la început costurile hardware erau preponderente în ansamblul cheltuielilor necesare realizării și exploatării sistemelor informatice, pe măsură ce tehnologiile de realizare a componentelor electronice s-au perfecționat, raportul costurilor hard-soft a început să se micșoreze, ajungând astăzi ca ponderea cheltuielilor de elaborare a programelor să constituie până la 90% din costul total al unui sistem de prelucrare automată a datelor. Creșterea continuă a costului scrierii de programe se datorează unor factori obiectivi; cel mai important este creșterea complexității programelor elaborate pentru problemele tot mai complicate rezolvate astăzi cu ajutorul calculatorului. Apoi, este necesară adaptarea programatorului la condițiile reale existente. Un alt factor ce contribuie la creșterea costului constă în schimbarea programului inițial, schimbare datorată și ea unor factori obiectivi. De asemenea, schimbarea configurației hardware poate duce la schimbări în program.

Astăzi se simte nevoia elaborării unor produse-program complexe, la care trebuie să participe mai mulți programatori. În general, produsele-program sunt clasificate în¹³³: simple (până la 1.000 de linii sursă, realizate în câteva luni), de complexitate medie (până la 10.000 de linii sursă și elaborate de 1-5 programatori, realizate în cel mult doi ani), complexe (până la 100.000 de linii sursă și elaborate de 5-20 de programatori în 2-3 ani) și foarte complexe (până la 1 milion de linii sursă, elaborate de 100-1000 programatori în câțiva ani). Prezentul program poate fi încadrat în categoria programelor de complexitate simplă.

Ciclul de realizare al unui sistem informatic este lung și presupune efectuarea unui volum ridicat de cheltuieli. Eficiența se definește¹³⁴ ca raportul între efectul util (ieșiri ale sistemului) și efortul făcut (intrări) pentru obținerea efectului respectiv. A estima cât mai corect costurile și eforturile necesare unui produs software este adesea o condiție esențială a competitivității. În vreme ce tehnica de calcul devine din ce în ce mai performantă datorită miniaturizării și ieftinirii, la elaborarea și întreținerea aplicațiilor software, în special a celor complexe, resursele consumate ajung să reprezinte 70-80% din costul total al sistemului, chiar cu posibilitatea de a ajunge la 90% în unele cazuri. Volumul mare al activității necesare pentru a dezvolta și întreține aplicațiile justifică interesul din ce în ce mai larg pentru asigurarea unei calități corespunzătoare a aplicațiilor precum și minimizarea costurilor de elaborare a acestora. Sunt utilizate metode și instrumente adecvate analizei amănunțite a cerințelor software și de elaborare sau testare a programelor, ce asigură un minim de erori admisibile și o productivitate cât mai mare.

Ciclul de viață al produsului software poate fi exprimat prin procentul de timp alocat fiecărei etape din procesul de dezvoltare. O repartitie clasică a acestor procente ar fi următoarea:

- Analiza cerințelor 7%
- Elaborarea specificațiilor 12%
- Proiectarea programelor 21%
- Implementarea programelor 20%
- Instalarea și testarea programelor 40%

Pentru a învinge concurența, produsele trebuie să coste cât mai puțin, să fie de calitate și să apară pe piață cât mai repede. Acest obiectiv se poate atinge doar printr-o abordare organizată, planificată a procesului de dezvoltare.

Produsul software poate fi evaluat în mod direct sau indirect. Prin evaluarea directă a produsului software se înțelege determinarea costurilor și a eforturilor asociate. Ea presupune calculul numărului liniilor de cod scrise, determinarea vitezei de execuție, a dimensiunii memoriei precum și a numărului defectelor raportate într-un anumit interval de timp. Evaluarea indirectă a produsului reprezintă de fapt o analiză a funcționalității, calității, complexității, eficienței, fiabilității, întreținerii și a altor caracteristici.

Metoda cea mai des utilizată în evaluarea produsului software este metoda evaluării dimensionale. Evaluarea dimensională reprezintă o estimare directă a acestuia precum și a procesului prin care acesta este dezvoltat. Cei mai importanți indicatori de evaluare sunt:

- Efortul, care estimează necesarul de resurse umane și se măsoară în programatori-pe-lună
- KLOC (Kilo lines of code) ce reprezintă numărul de mii de linii de cod

¹³³ [Vad85] Văduva I., Baltac V., Florescu V., Florică I., Jitaru M. -Ingineria programării, Editura Academiei RSR, București, 1985.

¹³⁴ [Oprean94] Oprean D., Racovițan D.M., Oprean V. -Informatică de gestiune și managerială, Editura Eurounion, Oradea, 1994, 173-181.

- Valoarea, care este exprimarea bănească a efortului
- Pagini de documentație
- Numărul de erori raportate de utilizatori într-o perioadă de timp (un an)
- Numărul de programatori care au lucrat la dezvoltarea produsului software

Din datele primare poate fi realizată o evaluare a productivității și una a calității după cum urmează:

Productivitatea = KLOC / Programatori / lună

Calitatea = Număr de erori / KLOC

Cost = Valoare / KLOC

Documentație = Pagini de documentație / KLOC

Utilizarea parametrilor dimensionali (KLOC, efort) este controversată și ei nu sunt universal acceptați ca reprezentând cea mai bună metodă de evaluare a procesului de dezvoltare software.

Pentru calculul eficienței economice a unei astfel de aplicații se poate folosi următoarea relație¹³⁵:

$E = N * f / [C_r + N * (C_a + C_e)]$ unde:

C_r reprezintă cheltuieli de elaborare a softului

C_a reprezintă cheltuieli de adaptare a softului

C_e reprezintă cheltuieli de exploatare a softului

N reprezintă numărul de utilizatori potențiali ai softului

F reprezintă efectele economice medii la beneficiari

Cheltuielile totale se constituie din cheltuieli de investiții, necesare analizei, proiectării, programării și implementării sistemului informatic și cheltuieli de exploatare care includ și cheltuielile făcute cu întreținerea lui. Acestea trebuie comparate cu venitul ce trebuie asigurat unui analist financiar în cazul în care nu s-ar pune problema elaborării unui astfel de sistem informatic. În unele cazuri, sistemul poate înlocui munca unei echipe întregi de analiști financiari.

Efortul este cuantificabil având în vedere că elementele de proiectare, programare și întreținere a aplicației se pot determina relativ ușor. Astfel se poate determina costul complet al aplicației până la instalarea completă și punerea în funcțiune și se poate estima costul întreținerii aplicației. Rămâne de determinat impactul economic produs de aplicație pentru proprietar. Pentru a fi utilă, aplicația trebuie să furnizeze informații pertinente, cât mai apropiate de momentul tranzacționării și în același timp să efectueze calcule complexe, cum ar fi: evaluarea indicatorilor referitori la portofolii de valori mobiliare, studiul evoluției și prognoza diferiților indicatori luând în calcul evoluțiile înregistrate până în prezent și evenimentele ce pot influența cursul acțiunilor.

Eficiența se pune și la nivelul utilizatorului. Din punctul de vedere al utilizatorului, programul are o interfață prietenoasă, pe bază de ferestre, permite utilizarea mouse-ului și prezintă caracteristicile oricărei aplicații Windows.

Necesitatea unor produse program performante cu costuri de exploatare și întreținere reduse a dus la abordarea problemelor privind fiabilitatea softului.

Fiabilitatea¹³⁶ se poate exprima cantitativ și calitativ.

Calitativ, fiabilitatea constituie capacitatea unui produs program de a îndeplini funcția prevăzută într-un timp dat, în condiții date și cu exactitatea cerută.

Din punct de vedere cantitativ, fiabilitatea este o caracteristică a produselor program exprimate prin probabilitatea ca acestea să îndeplinească funcția cerută cu anumite performanțe, fără erori în condiții date și într-un interval de timp dat. Măsura principală a fiabilității este probabilitatea funcționării produsului program fără căderi într-un interval de timp dat. Coeficientul numeric al fiabilității unui produs program este rata rulărilor cu succes din numărul total de rulări, aceasta fiind o măsură destul de bună în estimarea fiabilității.

¹³⁵ [Goron99] Goron S. –Proiectarea orientată obiect a produselor program, Editura Risoprint, Cluj-Napoca, 1999.

¹³⁶ [Goron99] Goron S. –Proiectarea orientată obiect a produselor program, Editura Risoprint, Cluj-Napoca, 1999.

5.7. INTEGRAREA SISTEMULUI DE ASISTARE A DECIZIILOR CU UN SISTEM EXPERT

Dezvoltarea în paralel a sistemelor de asistare a deciziei și sistemelor expert a condus la începutul anilor 1980 (1983-1984) la ideea integrării tehnologiei celor două tehnologii. Primele rezultate în domeniu au fost prezentate independent în cursul anilor 1985 de către Klein și Winston. Rezultatul major al acestor cercetări a fost concretizat în emergența unei noi structuri conceptuale de referință pentru instrumentele de dezvoltare a sistemelor interactive de asistare a deciziei. Structura conceptuală pentru instrumentele generatoare de DSS-expert este concepută de ei într-o manieră care să permită extinderea funcțiilor unui DSS astfel încât sistemul să fie capabil să furnizeze decidentului soluții pe care să le justifice. În cadrul acestei structuri regăsim funcții de modelare, prezentare a informațiilor, acces la instrumentele statistice de gestiune a datelor specifice unui DSS dar și funcții specifice SE: motorul de inferență și baza de cunoștințe. Motorul de inferență permite sistemului să simuleze comportamentul unui expert pornind de la datele și cunoștințele disponibile în timp de baza de cunoștințe alimentează motorul de inferență, într-un mod similar celui în care baza de date alimentează algoritmi statistici în cadrul DSS. Aceste sisteme au fost denumite DSS inteligente.

Esențială este integrarea componentei expert în structura DSS¹³⁷. Aceasta constă în trecerea controlului de la o funcțiune la alta în cadrul sistemului precum și la transferul informațiilor între funcțiuni (subsisteme). De exemplu în aplicațiile de asistare “inteligentă” a deciziei este esențial ca un model ce aparține de DSS să poată apela motorul de inferență care să utilizeze reguli în cadrul cărora se operează cu variabile folosite de către model sau regăsite în baza de date. Astfel, utilizatorul poate defini în baza de cunoștințe reguli ce utilizează orice variabilă a modelului asociat bazei de cunoștințe. Similar, expertul trebuie să poată returna controlul modelului sau generatorului de rapoarte pentru prezentarea rezultatelor.

Un exemplu de astfel de sistem ar putea fi operațional prin interfațarea unui sistem expert elaborat în Prolog cu un sistem de asistare elaborat în Visual Foxpro. Dacă cele două aplicații nu au un mediu comun de comunicare sau nu sunt scrise sub aceeași platformă de dezvoltare, este nevoie de generarea unui fișier de interfață prin care respectivele aplicații să comunice, așa cum este ilustrat în figura 44. Acest lucru are următorul inconvenient: informațiile cerute de sistemul expert nu sunt cunoscute în avans și acestea nu vor depinde decât în mică măsură de răspunsurile utilizatorului. Cea mai bună soluție de proiectare a unui DSS-expert este aceea de a utiliza un mediu comun între modulul expert și celelalte subsisteme componente astfel încât să nu fie nevoie de fișiere de interfață. Decidentul va utiliza un nume de variabilă în modulul expert, iar sistemul va returna valoarea corespunzătoare fie că aceasta este stocată în baza de date fie că este rezultatul unui calcul făcut în baza de modele.

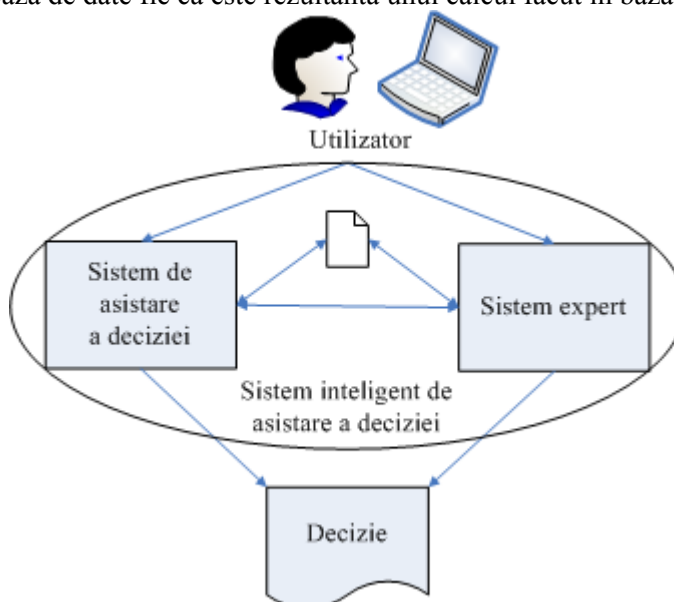


Fig. 44 Integrarea sistemului de asistare a deciziilor cu un sistem expert

¹³⁷ [Mărăcine98] Mărăcine V. –Decizii manageriale, Editura Economică, București, 1998

Instrumentele de tip DSS-Expert sunt utilizate frecvent în analiza și planificarea financiară, în domeniul bancar, al gestiunii portofoliilor și al activității de marketing. Utilizând instrumente de tip DSS-Expert de tipul PC-OPTRANS au fost dezvoltate numeroase sisteme destinate asistării decidenților în activități decizionale din sfera financiară.

CAPITOLUL 6. CONCEPEREA, PROIECTAREA ȘI IMPLEMENTAREA UNUI PROTOTIP DE GENERATOR DE SISTEME EXPERT ȘI A UNUI SISTEM EXPERT ÎN DOMENIUL BURSIER

6.1. PRODUSUL SOFTWARE UTILIZAT ÎN ELABORAREA GENERATORULUI DE SISTEME EXPERT

După cum s-a menționat în subcapitolul 4.3.2.2., pentru implementarea unui sistem expert se pot utiliza mai multe alternative:

- dezvoltarea sistemului expert pornind de la “zero”, utilizând fie limbaje de programare procedurale, fie declarative. Avantajul unui sistem expert dezvoltat cu ajutorul unui limbaj de programare este că asigură o flexibilitate în modalitatea de reprezentare a cunoștințelor, permițând astfel o modelare care să apropie cât mai mult sistemul de realitate. Dezavantajul unei astfel de abordări este dat de necesitatea implementării tuturor componentelor sistemului expert.
- utilizarea unui generator de sisteme expert, la care se atașează, în funcție de domeniul de aplicare al sistemului dorit, o bază de cunoștințe specializată în problema care se cere a fi rezolvată. Avantajul este dat de faptul că celelalte componente ale sistemului expert sunt asigurate de către generator. Această abordare poate crea anumite restricții datorită impunerii unui anumit mod de reprezentare a cunoașterii.

În prezenta lucrare ne-am propus să elaborăm un generator de sisteme expert care să permită pe de o parte experților să creeze baze de cunoștințe din diverse domenii, iar pe de altă parte utilizatorilor să beneficieze de rezultatele sistemelor expert astfel create.

Deoarece aplicația este realizată în PROLOG, vom face câteva referiri la aspecte privind acest limbaj de programare.

Ca și LISP, PROLOG este un limbaj cu fundament matematic. Bazele PROLOG-ului au fost puse de W.F. Clocksin și C.S. Mellish în lucrarea “Programming in PROLOG”¹³⁸. Începând cu 1982, a fost inclus în Japonia ca limbaj de bază în planul de dezvoltare a sistemelor de generația a 5-a, bazată pe procesarea cunoștințelor, program ambițios ce a avut succese. PROLOG are un câmp de aplicații foarte larg: baze de date relaționale, logica matematică, demonstrarea de teoreme, sisteme expert, rezolvarea de probleme abstracte sau ecuații simbolice, etc. PROLOG dispune de un motor de inferență de ordinul unu, deci acceptă reguli de producție cu variabile.

TURBO PROLOG este un mediu integrat de programare cu facilități de editare, compilare și execuție a programelor. Este realizat de firma Borland, pentru programe și aplicații ce folosesc limbajul de programare PROLOG. Prima versiune a acestui compilator a fost realizată în 1983. Acest mediu include un editor de texte, compilator, editor de legături, depanator simbolic și alte câteva funcționalități.

PROLOG s-a maturizat devenind un instrument puternic de programare a aplicațiilor de inteligență artificială, fiind implementat pe calculatoare VAX ca Quintus Edinburg Prolog și pe microcalculatoare ca Arity Prolog, ALS Prolog și altele. Dezvoltările recente au condus la realizarea unui puternic mediu de programare LPA Prolog, ce a condus la dezvoltarea a numeroase aplicații folosind acest mediu. În prezent, PROLOG s-a perfecționat prin crearea unui sistem foarte puternic, VISUAL PROLOG, care dispune de toate facilitățile unui mediu de programare vizual. Printre caracteristici se remarcă portabilitatea, modularitatea, flexibilitatea, predefinirea funcțiilor, execuția rapidă a aplicațiilor expert, depanare facilă, grafică interactivă, hipertext. VISUAL PROLOG a fost dezvoltat de firma daneză Prolog Development Center iar în prezent a ajuns la versiunea 6.2 lansată în noiembrie 2004.

Au fost înființate asociații puternice pentru promovarea și dezvoltarea științifică a acestui limbaj, printre care amintim: Prolog Management Group și Logic Programming Associates. Standardul oficial ISO Prolog a fost publicat în 1995.

PROLOG este un limbaj ideal pentru prototipizarea sistemelor expert, datorită calităților sale legate de viteza crescută de scriere, modificare și execuție a programelor, soliditate, flexibilitate, modularitate, trăsături ce duc la dezvoltări rapide de soluții parțiale în faza de proiectare și cu posibilități de iterare.

¹³⁸[Clocksin81] Clocksin W.F., Mellish C.C. –Programming in Prolog, Springer-Verlag, New York, 1981.

În PROLOG, soluțiile se obțin prin deducții logice plecând de la lucruri deja cunoscute. Un program PROLOG nu este o înșiruire de acțiuni de îndeplinit ci o colecție de fapte împreună cu regulile din care se pot trage concluzii din aceste fapte. PROLOG se bazează pe un subset al logicii predicatelor și oferă o sintaxă foarte apropiată de limbajul natural. Spre deosebire de LISP, specializat în definirea de funcții, PROLOG permite definirea și prelucrarea relațiilor. Un programator PROLOG va defini obiecte și relații, apoi va defini reguli despre care vor descrie când aceste relații vor fi adevărate.

Principial, fiind un limbaj descriptiv, în PROLOG există doar două tipuri de fraze: fapte și reguli, pe baza lor fiind capabil să efectueze operații de inferență logică. Aceste fraze sunt cunoscute în PROLOG sub numele de clauze. Un program PROLOG este constituit astfel dintr-o secvență de clauze. Clauza corespunde, prin referință la logica predicatelor de ordinul unu, unei propoziții simple sau complexe. O clauză se compune dintr-un predicat urmat de zero, unul sau mai multe argumente. Exemplu de predicat:

este(integer,integer,symbol)

unde *este* constituie numele predicatului, iar *integer, integer, symbol* sunt argumentele sale. Argumentele unui predicat pot fi date, constante sau nume de variabile.

Există două categorii de predicate:

-predefinite-fac parte din limbaj și constituie cuvinte rezervate

-definite de către utilizator –numele lor trebuie să înceapă cu o literă, urmată de orice secvență de litere, cifre sau simbolul “_”.

Aritatea unui predicat este numărul de argumente ale acestuia. Sunt admise predicate cu aceleași nume și cu arități diferite. Declarațiile predicatelor cu arități multiple trebuie grupate atât în secțiunea *predicates* cât și în secțiunea *clauses*.

Constantele reprezintă valori specificate explicit. Există două categorii de constante: atomii și numerele. Atomii sunt constante formate dintr-un șir de caractere; ca și predicatele, atomii se notează numai cu litere minuscule și nu pot conține spații. Dacă un atom este format din mai multe cuvinte, acestea se pot evidenția prin utilizarea semnului “_”:

identifica_entitate(integer,integer,integer)

Variabilele sunt elemente principale în programarea cu Prolog, fiind împreună cu constantele foarte des folosite pentru creșterea eficienței în programarea sistemelor expert. Variabilele sunt argumente ale căror conținut se stabilește sau se modifică în timpul prelucrării. Numele variabilelor trebuie să înceapă fie cu o literă majusculă, fie cu semnul “_”:

entitate(Numentitate,N,Nrlegatura)

Numeentitate, N și *Nrlegatura* reprezintă, în acest caz, numele a trei variabile.

PROLOG nu are instrucțiuni de atribuire, variabilele primesc valori prin potrivire cu constante din fapte sau reguli. Până când o variabilă primește o valoare, ea este liberă (free); când ea primește o valoare, ea devine legată (bound). Ea stă legată atâta timp cât este necesar pentru a obține o soluție a problemei. Apoi PROLOG o dezleagă, face backtracking și caută soluții alternative. Astfel se obține nu numai o soluție pentru problema dată ci toate soluțiile posibile.

Dacă dorim doar anumite informații de la o întrebare, se pot utiliza variabilele anonime pentru a neglija părțile care nu ne interesează. Variabila anonimă este reprezentată prin caracterul underscore “_”. Printr-o variabilă anonimă se semnalează doar prezența, fără a fi necesară cunoașterea conținutului său.

Clauzele care conțin un singur predicat sunt denumite clauze simple. O clauză poate fi compusă din mai multe clauze simple asociate prin conectorii logici *și* sau, devenind astfel o clauză compusă. Conectorul logic *și* se notează prin virgulă, iar conectorul sau prin punct și virgulă:

identifica_entitate(Cod_entitate,Nr_de_legatura,N):-

N=Nr_de_legatura+1;

legatura(Cod_entitate,Cod_caracterist,Cod_valoare,N),

identifica_caracterist(Cod_caracterist,Cod_valoare,DN),

DN="da",Np1=N+1,

identifica_entitate(Cod_entitate,Nr_de_legatura,Np1).

Faptele constau în descrierea realităților care afectează unul sau mai multe obiecte ale problemei de rezolvat, în descrierea unor situații sau în descrierea unor proprietăți ale unui obiect. Faptele se reprezintă în PROLOG prin aserțiuni. O aserțiune este o clauză simplă încheiată cu punct. Un fapt reprezintă o singură apariție fie a unei proprietăți a unui obiect, fie a unei relații între obiecte. Un fapt este de sine

stătător; PROLOG nu are nevoie să caute altceva pentru confirmarea unui fapt și faptul poate fi folosit ca bază pentru inferențe. Faptele, aspect important, pot fi deduse prin inferențe.

Constructorul din PROLOG care descrie ceea ce se poate deduce din alte informații este *regula*. O regulă este o proprietate sau o relație care se știe că este adevărată atunci când alte relații sunt adevărate. Regulile se reprezintă numai în forma inductivă. Cele două părți ale unei reguli –concluzia și premisele– sunt separate prin simbolul “:-” care are semnificația “dacă”. Atât concluzia cât și premisele sunt exprimate sub formă de clauze. Premisele unei reguli pot fi clauze (respectiv un nume de predicat standard sau definit de utilizator, cu sau fără argumente). Ca și faptele, o regulă se încheie întotdeauna cu punct. Punctul final nu face parte din regulă, el servește doar ca element lexical de specificare a sfârșitului clauzei respective.

Corpul regulii, care urmează după cap și simbolul if (“:-”), constă din unul sau mai multe subgoaluri. Fiecare subgoal este un apel la un alt predicat PROLOG. Când programul face acest apel, PROLOG testează predicatul apelat să vadă dacă poate fi adevărat. Odată ce subgoal-ul curent a fost satisfăcut (a fost găsit adevărat), se revine și procesul continuă cu următorul subgoal. Dacă procesul a ajuns cu succes la punct (care încheie regula), atunci regula a reușit. Pentru a utiliza cu succes o regulă, PROLOG trebuie să satisfacă toate subgoal-urile ei, creând o mulțime consistentă de legări de variabile. Dacă un subgoal eșuează (este găsit fals), procesul revine la subgoal-ul anterior și caută alte legări de variabile și apoi continuă. Acest mecanism se numește *backtracking*.

Comentariile se includ între /* și */. Alternativ, tot ceea ce apare pe o linie după caracterul % este comentariu.

Secțiunile de bază ale unui program PROLOG sunt:

-*secțiunea domains* –conține declarațiile domeniilor (tipurilor) folosite în program care nu sunt domenii standard. Tipurile standard sunt: char, integer, real, string și symbol. Domeniile definite de către utilizator au două scopuri utile: se pot da nume semnificative unor domenii, chiar dacă intern acestea sunt echivalente cu domenii care există deja și se pot declara domenii speciale, utile pentru a construi noi structuri de date, nedefinite de domeniile standard.

Dacă argumentele conțin informații aparținând domeniilor standard, este suficientă declararea lor în secțiunea predicates. În caz contrar, dacă se folosesc argumente de alte tipuri (definite de utilizator), acestea trebuie declarate obligatoriu în secțiunea *domains* înainte de a fi menționate în secțiunea *predicates*.

-*secțiunea predicates* –conține declarațiile predicatelor folosite și a tipurilor argumentelor folosite de aceste predicate. Argumentele pot fi de tipul standard sau de tipuri definite de utilizator în secțiunea *domains*

-*secțiunea clauses* –conține descrierea tuturor faptelor și regulilor ce se vor folosi. Clauzele ce descriu același predicat trebuie grupate astfel încât să apară una după alta în cadrul acestei secțiuni. În cazul predicatelor cu arități multiple, gruparea trebuie să se facă în funcție de numele predicatului și, în cadrul acestui criteriu, în funcție de aritate.

-*secțiunea goal* –conține întrebarea (goal) ce se adresează PROLOG-ului pentru a putea crea un program executabil. Secțiunea *goals* servește pentru definirea scopurilor externe. O facilitate importantă a limbajului PROLOG este că, în plus față de găsirea logică a răspunsurilor la întrebări, poate trata alternativele și să găsească toate soluțiile și nu numai una dintre ele. Un program PROLOG se poate executa interactiv, folosind așa numitele întrebări externe (external goals) parcurgând pașii următori:

-afișează pe ecran simbolul de așteptare (prompterul)

-citește o aserțiune sau o regulă de la terminal, o evaluează, afișează rezultatul pe ecran și reia ciclul.

Execuția programului constă în utilizarea definițiilor pentru a găsi o relație sau un obiect corespunzător unor specificații date. Dacă s-au găsit una sau mai multe soluții, acestea sunt afișate drept răspuns în fereastra de dialog; în caz contrar, apare mesajul “No solutions”.

Un program poate fi scris astfel încât să conțină și scopul (caz în care vorbim de scopuri interne). În acest caz, problema de rezolvat va fi unică (care corespunde scopului) iar soluția va fi și ea unică (chiar dacă există mai multe soluții posibile, va fi afișată prima dintre ele).

În funcție de tipul obiectivului evaluat (intern sau extern), se furnizează prima soluție găsită sau toate soluțiile. Uneori este nevoie de a găsi toate soluțiile atunci când în mod normal se furnizează doar prima soluție și invers, oprirea backtrackingului după găsirea primei soluții. Acestea se realizează folosind predicatele *fail* și *cut*.

Fail este un predicat fără argumente care semnifică întotdeauna eșec. Dacă este plasat în corpul unei clauze, semnificația de eșec obligă PROLOG-ul să caute în continuare găsirea de noi soluții. Răspunsul final (după găsirea tuturor soluțiilor) va fi întotdeauna “No” datorită prezenței acestui predicat care nu va permite satisfacerea acelei clauze niciodată..

Cut este tot un predicat fără argumente, simbolizat prin caracterul “!”. În poziția în care apare, oprește procesul de backtracking, întrerupând astfel căutarea de noi soluții. Plasat printre premisele unei reguli, el blochează revenirea atât la subscopurile care îl preced cât și la alte eventuale clauze care definesc același predicat.

Întreruperea forțată a execuției unui program se poate face folosind predicatul *exit*.

-*secțiunea constants* –permite declararea unor constante simbolice. Utilizarea constantelor simbolice contribuie la creșterea lizibilității programului. Există câteva restricții cu privire la utilizarea constantelor:

-numele de constante trebuie să fie unice

-definiția unei constante nu se poate referi la ea însăși

-numele constantelor se scriu în general cu litere mici (pentru a nu fi confundate cu nume de variabile)

-pot fi mai multe secțiuni de constante într-un program, dar constantele trebuie definite înainte de utilizarea lor.

-*secțiunea database* –uneori în timpul execuției unui program, dorim să actualizăm (schimbăm, ștergem sau să adăugăm) unele fapte cu care programul operează. Într-un astfel de caz, faptele constituie o bază de date dinamică sau internă. Predicatele care descriu baza de date internă se declară în secțiunea *database* și se utilizează în același mod în care se utilizează predicatele declarate în secțiunea *predicates*. Un program poate conține mai multe baze de date interne, descrise fiecare în câte o secțiune *database* distinctă.

Bazele de date sunt o extindere a noțiunii de fișier. PROLOG oferă posibilitatea de a lucra cu baze de date în memoria RAM. Declararea are loc în secțiunea *database*.

Assertz este un predicat specific pentru lucrul cu baze de date, permițând adăugarea acelui fapt la baza de date, din acel moment faptul aparținând bazei.

assertz(comentariu_entitate(Tip))

Faptele pot fi șterse folosind *retract* sau *retractall*:

retractall(comentariu_entitate(_))

Predicatul *consult* citește fapte dintr-un fișier și le inserează în baza de fapte:

consult(Nume_fis)

Predicatul *save* salvează într-un fișier toate faptele relative la predicatele din baza de date implicită:

Save(Nume_fis).

6.2. APLICAȚIA “EXPERT GENERATOR” –ARHITECTURĂ ȘI CARACTERISTICI GENERALE

Datorită backtracking-ului automat, motorul de inferență al Prolog-ului permite o dezvoltare ușoară a aplicațiilor bazate pe acest limbaj. Aplicația “Expert Generator” dezvoltată în acest sens cuprinde: un modul de achiziție și editare de cunoștințe, un motor de inferență ce acționează asupra uneia dintre bazele de cunoștințe astfel introduse precum și o interfață cu utilizatorul ce permite atât crearea bazelor de cunoștințe cât și identificarea pe baza cunoștințelor introduse.

În figura 45 este ilustrată arhitectura aplicației “Expert Generator”.

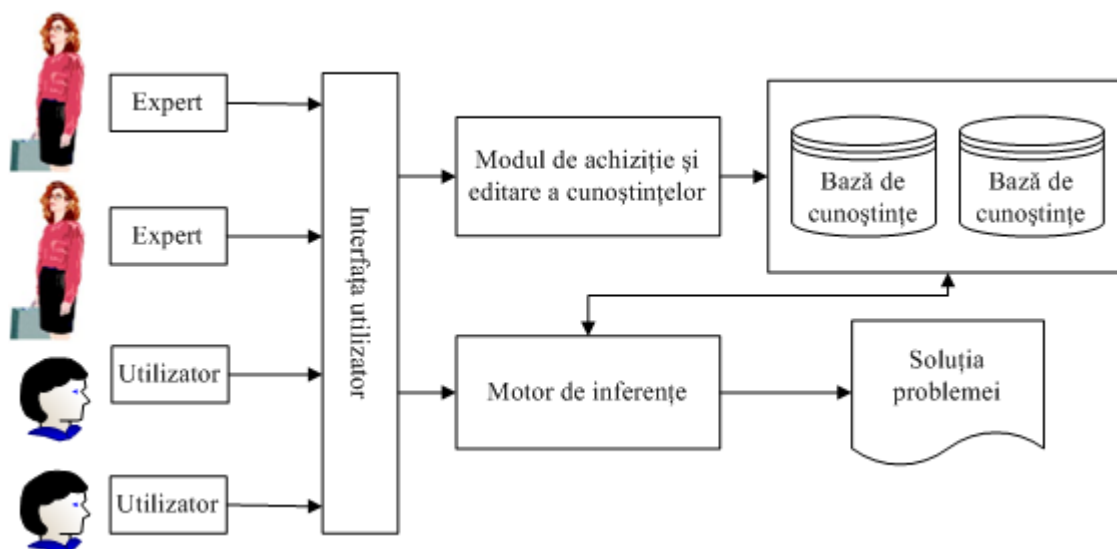
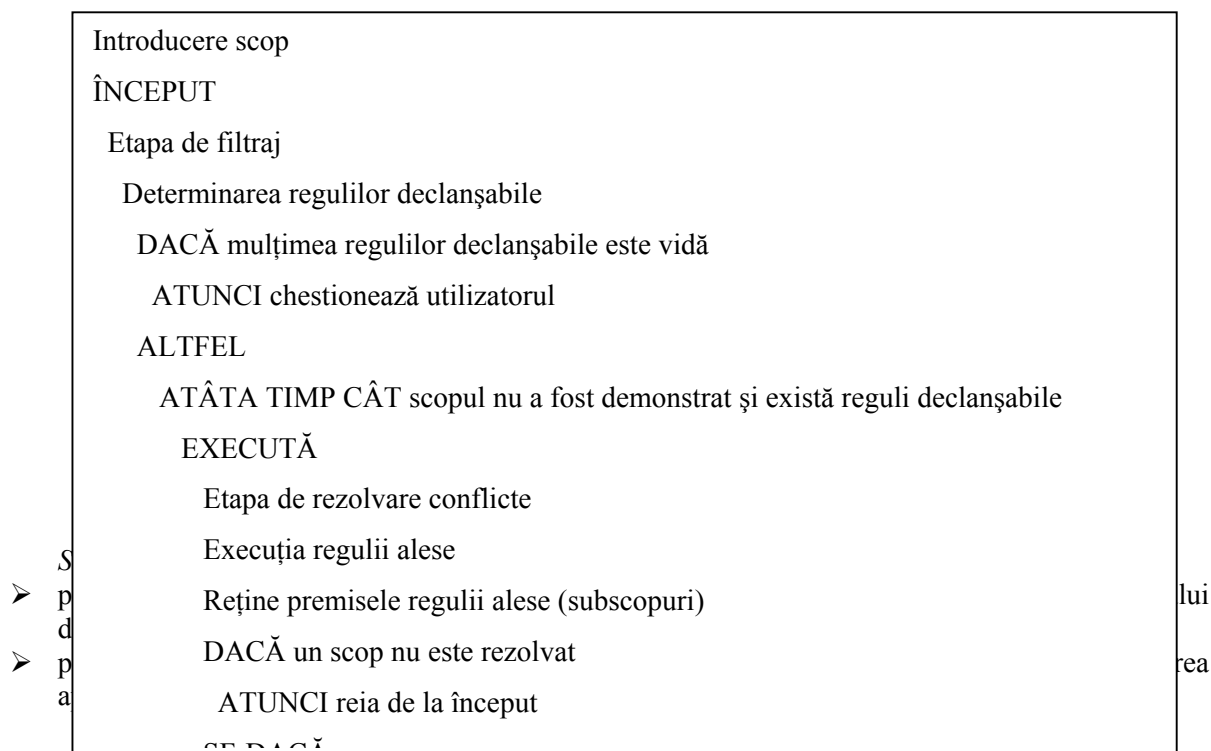


Fig.45 Arhitectura aplicației “Expert Generator”

Aplicația fiind elaborată în Prolog, *raționamentul* folosit este cel inductiv, motorul de inferență plecând de la un anumit scop (obiectiv) pe care îl descompune în subprobleme până când obține problemele primitive (faptele dovedite sau interogabile). Acest mod de raționament este dirijat de scop în sensul găsirii faptelor care permit atingerea acestuia (invers decât raționamentul deductiv). Regulile selecționate sunt acelea care au în partea de concluzii (dreapta) scopul specificat inițial. Premisele acestora vor deveni subscopuri ce urmează să fie demonstrate. Acest proces se repetă până când toate subscopurile obținute sunt demonstrate sau în urma etapei de filtraj mulțimea regulilor declanșabile este vidă (în caz de eșec).

Algoritmul inductiv utilizat în Prolog poate fi sintetizat astfel¹³⁹:



Beneficiarul principal al acestei aplicații poate fi orice utilizator care dorește consultații la nivel de expert într-un anumit domeniu.

Interfața aplicației este prietenoasă și se prezintă sub forma unui sistem de meniuri apelabile prin intermediul unor taste funcționale.

¹³⁹ [Zaharie99] Zaharie D., Năstase P. și alții –Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999

6.3. BAZA DE CUNOȘTINȚE A APLICAȚIEI

Regulile de producție reprezintă cea mai folosită metodă de reprezentare a cunoștințelor într-un sistem expert. Această metodă are avantajul simplității faptelor și regulilor și a naturaleții transpunerii cunoașterii, pretându-se mai ales în domeniile în care cunoașterea este factuală.

Aplicația “Expert Generator” crează baze de cunoștințe organizate sub forma unor fișiere text, fișiere având extensia .cun.

Baza de cunoștințe a acestei aplicații este structurată astfel:

Database

```
/*pentru precizarea tipului obiectului de identificat*/
comentariu_obiect(symbol)
```

```
/*ENTITATE <nume_entitate> <cod_entitate> <nrdecun> */
entitate(symbol,integer,integer)
```

```
/*CARACTERISTICI <nume_caracteristica> */
/*<cod_caracteristica> <mesaj_inainte> <mesaj_dupa>*/
caracter(symbol,integer,symbol,symbol)
```

```
/*EXCLUSIV <cod_caracteristica>*/
exclusiv(integer)
```

```
/*VALOARE <cod_caracteristica> <nume_valoare> <cod_valoare>*/
valoare(integer,symbol,integer)
```

```
/*CUNOȘTINȚE<cod_entitate><cod_caracteristica>*/
/*<cod_valoare><cod_cunostinta>*/
cun(integer,integer,integer,integer)
```

Între entități și cunoștințe, caracteristici și cunoștințe, valori și cunoștințe, caracteristici și valori există legături 1-n. Ilustrarea acestor legături se face în figura 46.

Exemplificarea acestor noțiuni se face în subcapitolul 6.5., odată cu crearea unui sistem expert în domeniul bursier folosind aplicația “Expert Generator”.

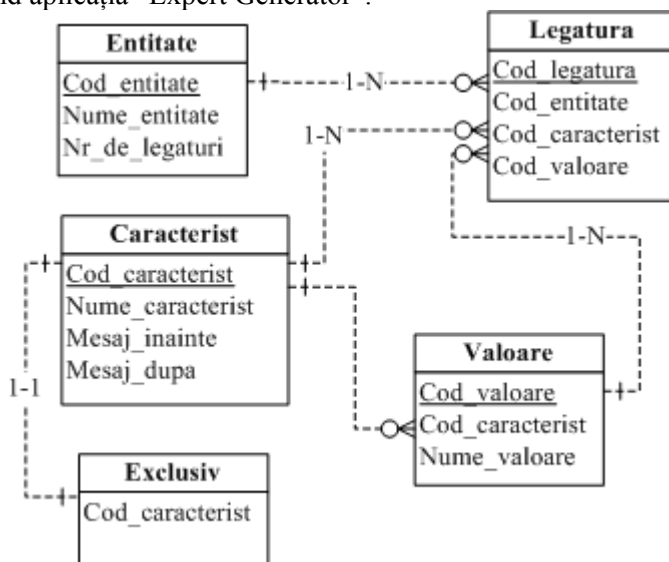


Fig.46 Structura bazei de cunoștințe a aplicației “Expert Generator”

Achiziția cunoștințelor se face printr-o metodă semi-automată, folosind un sistem de întrebări și răspunsuri, cu ajutorul cărora expertul crează baza de cunoștințe. La întrebările sistemului, expertul răspunde prin selecția unei opțiuni din meniul afișat pe ecran. Pe baza răspunsurilor la aceste întrebări, sistemul constituie o bază de cunoștințe ce va putea fi încărcată de diverși utilizatori în vederea consultării ei.

Baza de cunoștințe într-un sistem expert nu este statică, ea trebuie în permanență actualizată cu cunoștințele noi apărute. De aceea, sistemul permite adăugarea, modificarea și ștergerea de reguli de producție din baza de cunoștințe.

6.4. IMPLEMENTAREA APLICAȚIEI

Aplicația “Expert Generator” a fost implementată în mediul de programare “Turbo Prolog”. Am ales această soluție datorită posibilității de a utiliza motorul de inferență oferit de Prolog. În cazurile în care se dorește utilizarea comercială a acestei aplicații, ea poate fi portată spre mediul de programare Visual Prolog.

Codul sursă al aplicației “Expert Generator” este prezentat în anexa 17.

Un marele avantaj al generatorului de sisteme expert elaborat este generalitatea, adică posibilitatea de a fi utilizat într-o diversitate de domenii, atât de experți cât și de diverși utilizatori care doresc consultații la nivel de expert, fără a fi nevoie de intervenții în aplicație, intervenții ce necesită cunoștințe de programare.

6.5. UTILIZAREA APLICAȚIEI “EXPERT GENERATOR” PENTRU CREAREA UNUI SISTEM EXPERT ÎN DOMENIUL BURSIER

În exemplul pe care îl vom considera în continuare, care utilizează generatorul de sisteme expert dezvoltat, am folosit următoarele informații în structura bazei de cunoștințe:

Entități:

Acțiune preferențială
Acțiune obișnuită
Acțiune de aport
Acțiune cu dividend prioritar
Obligațiune clasică
Obligațiune cu dobânda variabilă
Obligațiune cu cupon zero
Obligațiune cu durată variabilă

Caracteristici:

Durată
Dividend
Risc
Venit
Dobândă
Vot
Contravaloare

Valori:

Durată - nelimitată
 - limitată
 - variabilă
Dividend - limitat

Risc	- prioritar - scăzut - mediu - ridicat
Venit	- sigur - nesigur - variabil - fix
Dobânda	- fixă - variabilă - zero
Vot	- da - nu
Contravaloare	- bunuri - bani

Exclusiv Valoarea acesteia este “nu” pentru caracteristici ce pot fi îndeplinite cumulativ și “da” în caz contrar.

Durata	-da
Dividend	-nu
Risc	-da
Venit	-nu
Dobanda	-da
Vot	-da
Contravaloare	-da

Legătura

Acțiune preferențială:	-durata(variabilă), dividend(limitat), vot(nu), contravaloare(bani), venit(variabil), risc(mediu).
Acțiune obișnuită:	-durata(nelimitată), dividend(limitat), vot(da), contravaloare(bani), venit(variabil), venit(nesigur), risc(ridicat).
Acțiune de aport:	-durata(nelimitată), dividend(limitat), vot(da), contravaloare(bunuri), venit(nesigur), venit(variabil), risc(ridicat).
Acțiune cu dividend prioritar:	-durata(nelimitată), dividend(prioritar), vot(nu), contravaloare(bani),

venit(nesigur),
venit(variabil),
risc(mediu).

Obligațiune clasică: -durata(limitată),
dobinda(fixă),
vot(nu),
contravaloare(bani),
venit(sigur),
venit(fix),
risc(scăzut).

Obligațiune cu dobânda variabilă: -durata(limitată),
dobânda(variabilă),
vot(nu),
contravaloare(bani),
venit(variabil),
risc(scăzut).

Obligațiune cu cupon zero: -durata(limitată),
dobânda(zero),
vot(nu),
contravaloare(bani),
venit(sigur),
venit(fix),
risc(scăzut).

Obligațiune cu durata variabilă: -durata(variabilă),
dobinda(fixă),
vot(nu),
contravaloare(bani),
venit(sigur),
venit(fix),
risc(scăzut).

O parte din fazele creării unei baze de cunoștințe în domeniul bursier având structura bazei de cunoștințe descrisă mai sus, precum și faza de identificare a produsului financiar dorit sunt exemplificate în anexa 18.

Baza de cunoștințe astfel creată poate fi consultată în anexa 19.

De asemenea, în anexa 20 prezentăm și un sistem expert independent elaborat în Prolog, având la bază aceleași cunoștințe care servesc la exemplificarea aplicației “Expert Generator”. Prin rularea comparativă a celor două sisteme obținem aceleași rezultate.

6.6. APRECIERI PRIVIND EFICIENȚA SISTEMELOR EXPERT

Orice produs este caracterizat prin durata sa de viață. În lumea informaticii, principala caracteristică a produselor program este durata de viață deosebit de scurtă. Cauza o constituie dinamica foarte mare a domeniului tehnologiei informației, evoluția tehnologiei și numărul mare de specialiști implicați în această industrie. Până în prezent nu există un punct de vedere comun în privința metodelor de cuantificare în vederea aprecierii unui sistem expert¹⁴⁰. Economic vorbind, eficiența unui sistem

¹⁴⁰ [Goron00]Goron S., Joldeș R., Ileană I. –Inițiere în inteligența artificială, Editura Risoprint, Cluj-Napoca, 2000.

expert, analog cu cea a unui produs program, poate fi realizată prin compararea efectelor economice realizate cu costul resurselor utilizate.

Considerăm că evaluarea unui sistem expert vizează, de asemenea, calitatea și utilitatea acestuia. Pentru a evalua rezultatele oferite de sistemul expert în comparație cu experții umani, propunem generarea unui set de teste care să cuprindă cazuri obișnuite, simple și deosebit de grele, cazuri ce vor fi oferite spre rezolvare atât unei grupe de experți cât și sistemului implementat. Evaluarea rezultatelor va fi făcută de o altă grupă de experți, fără ca aceasta să știe de unde provin soluțiile. Propunem următoarea formulă pentru aprecierea gradului de eficiență al sistemului:

$$P_i = 3 \cdot N_E + 2 \cdot N_B + 1 \cdot N_A + 0 \cdot N_N$$

unde: P_i este performanța expertului i

N_E este numărul de soluții catalogate ca excelente

N_B este numărul de soluții catalogate ca bune

N_A este numărul de soluții catalogate ca acceptabile

N_N este numărul de soluții catalogate ca inacceptabile.

Numărul de soluții din fiecare caz a fost ponderat cu valorile 3, 2, 1 și 0 aferent cazurilor excelente, bune, acceptabile și inacceptabile. Gradul de performanță al fiecărui expert, grad măsurat prin indicatorul P_i va fi comparat cu gradul de performanță al sistemului expert, calculat prin aceeași formulă.

În general produsele de inteligență artificială, în special sistemele expert, sunt mai scumpe decât alte categorii de produse informatice de pe piață datorită faptului că:

- piața unui produs gen sistem de operare este foarte mare în comparație cu cea a unui sistem expert, aceasta din urmă fiind punctuală, dedicată unor probleme specifice, deci piața sa este restrânsă;
- implementarea unui astfel de sistem necesită costuri mari (crearea bazelor de cunoștințe, antrenarea utilizatorilor sunt operații cu costuri ridicate);
- sistemele expert sunt rodul unor eforturi deosebite, implică muncă specializată, efectuată de persoane cu o pregătire și calificare ridicată. Aceasta implică cheltuieli salariale ridicate.

Avantajele utilizării unui sistem expert într-o întreprindere rezidă în primul rând din *productivitatea ridicată* a sistemului. Considerăm că avantajele se pot urmări în timp astfel:

-pe termen scurt –sistemul trebuie să rezolve sarcina mai repede decât specialistul. Pentru un sistem expert de evaluarea a riscului clientului, timpul necesar pentru studierea unui dosar se scurtează de la o oră la 10 minute. Pentru controlul dosarelor privind analiza bilanțului, timpul ar fi de 7 ori mai mic, aceeași eficiență manifestându-se și în cazul controlului dosarelor privind securitatea socială.

Scurtarea termenului se poate referi și la creșterea calificării sau accesul unor persoane la sarcini pe care acestea erau incapabile să le desfășoare până atunci. Rareori un foarte bun expert este înlocuit în totalitate de un sistem expert dar problema se pune altfel pentru expertize medii. În multe cazuri, expertiza medie realizată cu ajutorul unui sistem expert se poate face de către persoane cu o calificare inferioară.

-pe termen lung –câștigurile pe termen lung se referă în principiu la gestiunea expertizei și calitatea deciziilor. În majoritatea cazurilor se poate afirma că introducerea unui sistem expert conduce la ameliorarea calității deciziilor. Sistemele expert deschid largi posibilități raționamentelor, prin explorarea unui număr de variante mai mare decât cele pe care le ia în calcul mintea umană. În activitățile de planificare și concepție, acest factor de ameliorare a deciziilor are o importanță deosebită. Un alt aspect important prin care sistemele expert aduc avantaje deosebite considerăm că este cel al *structurării cunoștințelor*. Realizarea sistemului expert conduce la modelarea și structurarea cunoștințelor specialiștilor, acest avantaj având acțiune directă asupra formării personalului. Alt avantaj ține de actualizarea cunoștințelor: dacă acestea sunt structurate, este ușor de făcut.

Avantajele oferite de un sistem expert pot fi foarte variate, de la cele mai subtile până la avantaje majore. Se poate aminti accidentul de la Three-Miles Island din 1979, unde, datorită blocării unei valve de scurgere a apei din reactor în poziția deschis și a erorii operatorilor, centrala a trecut pe lângă un accident de tip Cernobîl. Aici, operatorii umani, depășiți de numărul de puncte sinoptice (peste 1300 de cadrane și indicații luminoase) au acționat greșit, putând arunca centrala în aer. Sistemele automate au diagnosticat corect problema și au reușit să diminueze efectele accidentului.

Datorită acestor considerente, apreciem că sistemele expert reprezintă un mare avantaj în buna desfășurare a activității oricărei firme.

CAPITOLUL 7

CONTRIBUȚII PROPRII

Lucrarea prezintă o abordare a unei tematici pluridisciplinare din două perspective: economică și tehnică. Tematica aleasă se găsește la intersecția mai multor domenii de cercetare: domeniul financiar-bancar, în particular cel al pieței de capital, cel al teoriei deciziei în condiții de risc și incertitudine și în principal domeniul inteligenței artificiale. Autorul studiază tehnologia de elaborare a sistemelor de asistare a deciziilor și a sistemelor expert și impactul pe care inteligența artificială îl are asupra domeniului cercetat. Lucrarea de față se constituie deci într-un studiu sistematic al metodelor și tehnicilor sistemelor inteligente în domeniul pieței de capital.

Cercetarea întreprinsă cuprinde două părți: una teoretică, în care este abordată științific tematica propusă spre cercetare și una practică, ce este concretizată prin proiectarea și implementarea a două aplicații informatice bazate pe domeniul studiat, acestea constituind contribuția esențială a autorului la prezenta teză de doctorat.

Contribuțiile personale ale autorului, derulate pe parcursul lucrării, pot fi sintetizate astfel:

În *primul capitol*, autorul face o introducere în problematica cercetării și delimitează obiectivele care constituie scopul acestei lucrări.

Capitolul doi intitulat “Stadiul cunoașterii” realizează o trecere în revistă a principalelor concepte din domeniile studiate. Autorul vine cu o abordare critică și își exprimă propriile opinii privind conceptele din domeniu și formulează o serie de concluzii, reușind să creeze cadrul necesar abordării aprofundate a noțiunilor legate de piața de capital și de inteligența artificială, aspecte ce vor fi detaliate în capitolele trei și patru.

În *capitolul trei* sunt expuse cele trei mari teorii care stau la baza analizei pieței de capital. În cadrul analizei fundamentale, autorul face interpretarea indicatorilor prezentați, indicatori ce vor fi apoi utilizați în cadrul aplicației de asistare a deciziei de investiție din cadrul capitolului cinci. Autorul surprinde în continuare importanța analizei tehnice și a teoriei piețelor eficiente, ca și instrumente ce vin să completeze informațiile oferite de indicatorii analizei fundamentale. La fiecare subcapitol, autorul își prezintă propriile puncte de vedere, sub forma unor concluzii.

Capitolul patru constituie punctul forte din punct de vedere teoretic al prezentei lucrări. Autorul îmbină definițiile, formulările riguroase, cu stilul critic. Remarcăm comparațiile deosebit de sugestive precum și bogata ilustrare figurativă a conceptelor cu care autorul sporește atractivitatea, conducând spre un conținut ușor de înțeles și totuși dens informațional.

În cadrul acestui capitol, autorul își prezintă propriile puncte de vedere privind principalele categorii de sisteme inteligente. Sistemele de asistare a deciziilor și sistemele expert sunt prezentate în detaliu, ele constituind baza teoretică pentru aplicațiile practice. Autorul vine cu argumente proprii în favoarea susținerii acestora ca instrumente necesare în procesul decizional de investiție.

Autorul descrie funcționalitatea celor mai cunoscute sisteme expert ce sunt utilizate pe plan mondial în domeniul pieței de capital, dovedind prin aceasta necesitatea utilizării unor instrumente de acest tip în domeniul bursier.

Interpretările proprii finalizate cu concluziile de la sfârșitul fiecărui subcapitol întregesc aspectele teoretice prezentate.

Autorul nu se limitează la studiul teoretic al domeniului de cercetare. În consecință, teza are și un profund caracter practic, elementele teoretice prezentate fiind exemplificate în capitolele 5 și 6 prin proiectarea și implementarea unui sistem de asistare a deciziilor și a unui generator de sisteme expert în acest domeniu. Dezvoltarea acestor sisteme dovedesc caracterul practic pe care autorul îl dă tezei de doctorat.

În *capitolul cinci* autorul concepe un sistem de asistare a deciziilor în domeniul valorilor mobiliare, având ca suport teoretic capitolul 3 și subcapitolul 4.2. Sistemul proiectat și implementat reprezintă o contribuție importantă, constituind un instrument de lucru cu largi funcționalități. Aplicația a fost dezvoltată în mediul Visual FoxPro 8.0 (unul din cele mai moderne și larg răspândite medii de dezvoltare a aplicațiilor) și constituie un instrument ce poate fi folosit în procesul de analiză în vederea luării celei mai bune decizii de investiție. Această aplicație se dovedește a fi deosebit de utilă analiștilor, brokerilor, traderilor, agenților de valori mobiliare și în general oricărui investitor în vederea îmbunătățirii calității deciziilor. Sistemul oferă o serie de situații de ieșire, care se constituie în sinteze și recomandări ce servesc analistului drept suport în vederea deciziei de investiție pe care urmează să o ia.

O contribuție originală și în același timp deosebită o reprezintă *capitolul șase* în care autorul reușește să proiecteze și să implementeze un prototip de generator de sisteme expert. Avantajul acestui generator este generalitatea, adică posibilitatea de a fi utilizat într-o diversitate de domenii. Generatorul de sisteme expert proiectat permite atât crearea de multiple baze de cunoștințe cât și identificarea unei anumite entități pe baza cunoștințelor astfel introduse de către utilizatorii care doresc consultații la nivel de expert. Achiziția cunoștințelor este facilă, pe baza unui sistem de întrebări și răspunsuri din partea expertului prin selecția unei opțiuni din meniul afișat pe ecran. Funcționalitatea acestui generator este demonstrată printr-o exemplificare axată pe domeniul pieței de capital, sistemul expert creat cu ajutorul acestui prototip realizând identificarea unui produs financiar pe baza caracteristicilor sale. În paralel este dezvoltat și un sistem expert independent, implementat tot în Prolog, având la bază aceleași cunoștințe și care oferă aceleași rezultate ca și sistemul realizat folosind generatorul.

Capitolul opt cuprinde concluziile finale și propunerile desprinse din prezenta cercetare. Autorul formulează 12 concluzii de importanță majoră privind tematica studiată. Cele 4 direcții viitoare de cercetare stabilite de către autor încheie teza de doctorat.

Cercetările întreprinse au fost publicate, majoritatea ca unic autor, în cadrul mai multor cărți de specialitate și articole, prin participarea la sesiuni științifice naționale și internaționale. În acest sens se pot cita următoarele referințe: [Bologa03], [Bologa02c], [Bologa01a], [Bologa01b], [Bologa00], [Bologa99], [Bologa98], [Bologa97a], [Bologa97b], [Bologa96a], [Bologa95b] în calitate de unic sau prim autor și [Goron02], [Goron03] în calitate de coautor. În cadrul activității de cercetare se pot menționa referințele [Racovițan00] și [Rusu01], în calitate de coautor la rapoartele anuale de cercetare ale unor contracte de cercetare CNCIS. Activitatea de laborator s-a concretizat într-o culegere de exerciții și probleme referită prin [Nițchi03b]. Tot aici se pot remarca premiile obținute de autor în domeniul cercetării din acest domeniu la diverse manifestări științifice printre care se poate menționa referința [Bologa95a] încununată cu premiul I, dar și concursurile de specialitate cum ar fi "Cel mai bun programator", concurs organizat de Academia de Studii Economice București 4-7 mai 1995 (premiul II la secțiunea avansați) și participarea la concursul internațional "The 1995-1996 ACM International Collegiate Programming Contest -Eastern European Regional Contest- Bucharest, 2-5 November 1995".

CAPITOLUL 8

CONCLUZII FINALE ȘI PROPUNERI

8.1. CONCLUZII

1. Dezvoltarea tehnologiilor IT și îndeosebi a Internetului a contribuit la crearea unor mișcări speculative de capitaluri. Pe de o parte, economiile moderne aflate sub imperiul globalizării piețelor au adus pentru investitorii pe piețele de capital posibilitatea constituirii unor portofolii de active financiare (acțiuni, obligațiuni, bonuri de tezaur, instrumente derivate, alte titluri de valoare) diversificate din punct de vedere structural, (al componenței în total portofoliu) sectorial (ramurile economiilor naționale unde au fost efectuate plasamentele), și geografic (țările în care s-au activat investițiile). Această globalizare a fost posibilă datorită dezvoltării considerabile a tehnologiei IT, într-o convergență deplină cu dezvoltarea piețelor monetare ale țărilor cuprinse în portofoliu. Pe de altă parte, piețele de capital ale țărilor foste comuniste au oferit și oferă oportunități deosebite în ceea ce privește îndeosebi investițiile în acțiuni.

2. Volatilitatea și incertitudinea se extind cu repeziciune în lumea globală. Volatilitatea piețelor financiare accentuează necesitatea studiului riscurilor, cu atât mai mult cu cât fenomenul se face simțit tot mai puternic în economia globală. Pe fondul reducerii considerabile a costului de curtaș al piețelor, mișcările speculative de capitaluri măresc probabilitatea de supraevaluare a prețului unor active financiare pe piețele emergente. Combinarea analizei fundamentale și tehnice cu tehnologia oferită de inteligența artificială conduc spre utilizarea unor instrumente ce permit evaluarea rapidă a valorilor mobiliare care se pot constitui la un moment dat în investiții. Aceste instrumente constituie un suport de decizie absolut necesar în mâna oricărui investitor sofisticat pe piața de capital.

3. Piața de capital este un sistem complex, iar evoluția prețurilor valorilor mobiliare este afectată de o multitudine de factori economici și psihologici. Devine din ce în ce mai greu pentru un investitor să colecteze, filtreze, evalueze și să utilizeze masa mare de informații disponibilă în domeniul pieței de capital în vederea luării unei decizii de investiție. Creșterea puternică a World Wide Web-ului a creat oportunități majore, punând la dispoziția decidenților un volum imens de informații, în cea mai mare parte nestructurate. Aceste depozite de date sunt la dispoziția celor care au nevoie de ele și permit analize economice din ce în ce mai complexe, care să folosească întreaga valoare pe care o posedă datele colectate.

Tehnologiile IT și îndeosebi instrumentele de asistare a deciziei se găsesc într-o fază de dezvoltare puternică. Managerii care înțeleg să utilizeze aceste tehnologii beneficiază de avantaje majore în competiția impusă de economia de piață. Ei trebuie să valorifice informațiile în decizii manageriale inteligente, atât la nivel strategic cât și la nivel tactic. Din cauza timpului scurt de luare a deciziilor, tot mai multe firme vor apela la sisteme inteligente care să asiste managerii în luarea deciziilor importante. În acest proces instrumentele de analiză au rolul principal.

4. Fără îndoială că pentru majoritatea IMM-urilor din România sistemele având la bază inteligența artificială constituie încă instrumente decizionale costisitoare. O decizie eficientă nu este însă doar una a cărei adoptare și implementare implică cele mai reduse costuri ci și una complet și complex fundamentată într-un interval de timp cât mai scurt și care are efecte pozitive asupra sistemului sau aspectului pe care îl vizează. În atingerea acestor obiective ale activității decizionale, rolul sistemelor inteligente și îndeosebi a sistemelor expert nu poate fi contestat.

În ceea ce privește construirea efectivă a sistemelor expert, practicienii și cercetătorii în domeniu au concluzionat faptul că o construcție facilă, rapidă și la un cost redus a unor astfel de sisteme necesită un mediu de dezvoltare specializat. Astfel s-a născut ideea construirii unor generatoare de sisteme expert, constând într-un mediu software care facilitează conceperea de către utilizatori a unor sisteme expert

proprii și permit realizarea funcțiilor sistemelor expert. Acest lucru i-a făcut pe cercetătorii în domeniu să își concentreze eforturile asupra dezvoltării unor instrumente software posibil de utilizat direct de către beneficiari în urma unei pregătiri minimale, instrumente denumite generatoare de sisteme expert. Acestea furnizează resurse pentru dezvoltarea aplicațiilor și automatizează integrarea dintre aceste resurse. Capacitatea de adaptare la necesitățile utilizatorului este foarte mare. Generatoarele de sisteme expert cunosc deci o dezvoltare și perfecționare continuă. Performanțele lor s-au îmbunătățit în paralel cu ale tehnicii de calcul utilizate și sub influența celorlalte domenii ale inteligenței artificiale.

5. Considerăm însă că cel mai puternic impact al sistemelor expert și al inteligenței artificiale în general asupra tuturor categoriilor de întreprinderi din mediul de afaceri românesc îl poate reprezenta într-o primă fază contribuția adusă de aceasta la consolidarea conceptului de organizație bazată pe cunoștințe și la definirea unor strategii coerente pentru managementul cunoștințelor economice în cadrul acestor organizații.

De asemenea, una din concluziile care se poate trage din implementarea unui sistem expert este marele avantaj care apare din structurarea datelor firmelor. Spre deosebire de un sistem informatic clasic, într-un sistem expert se remarcă lizibilitatea bazei de cunoștințe. Regulile se pot scrie într-o manieră care să conducă la o bază de cunoștințe lizibilă și să asigure o structurare coerentă a cunoștințelor din domeniu. Crearea unui astfel de sistem este o ocazie foarte bună pentru înțelegerea procesului investițional și a locului firmei în cadrul pieței.

6. În ciuda acceptării entuziaste și răspândirii sistemelor expert în industrie și comerț, există și unele păreri negative asupra utilizării acestei tehnologii și a rezultatelor ei. Aceste percepții negative se bazează pe următoarele argumente: tehnologia nu a făcut față trecerii de la stadiul cercetărilor de laborator la cel al aplicațiilor reale, s-a dovedit a fi prea dificil de aplicat, tehnologia nu a răspuns așteptărilor și cerințelor promise, iar utilizatorii au fost dezamăgiți. Dacă tehnologia ar fi fost atât de bună pe cât s-a promis, companiile ar fi putut s-o adopte mai repede și cu efort/cost mai mic. Deși se pare că există motive pentru fiecare dintre aceste argumente, putem aprecia pe baza unei analize mai aprofundate că nici unul din argumente nu este valid. Tehnologia nu a eșuat la trecerea din laborator în industrie, iar insuccesele s-au datorat fie imaturității, la un anumit moment, a soluțiilor propuse, fie încercării de aplicare izolată a acesteia. Utilizarea actuală a sistemelor inteligente este de cele mai multe ori limitată mai ales datorită lipsei de integrare între diversele componente program decât datorită incapacității componentelor bazate pe cunoștințe de a rezolva problema. Cu toate acestea, rata de difuzie a sistemelor inteligente în aplicațiile practice se dovedește a fi printre cele mai ridicate dintre cele ale răspândirii unei noi tehnologii. În mod evident, sistemele inteligente nu au reușit să creeze sistemul promis în anii '60-'70 care să imite total omul sau care să treacă testul Turing, dar avantajele prezentate mai sus au demonstrat utilitatea și eficacitatea acestora.

7. În ultimii ani, cea mai mare creștere a productivității prin aplicarea sistemelor expert a fost obținută prin dezvoltarea și utilizarea sistemelor orientate pe problemă. Deoarece aceste sisteme au fost proiectate în vederea generării de soluții ale unui tip specific de probleme, creatorii lor au putut adopta o arhitectură de rezolvare a problemelor standard cu reprezentări de cunoștințe și tehnici de control adecvate. Aceasta a permis creșterea eficienței sistemului și simplificarea sarcinii de transferare a cunoștințelor pasive acumulate, într-un cadru de raționament activ.

8. Cunoașterea tehnologiei de realizare a sistemelor de asistare a deciziilor și a sistemelor expert permite utilizarea complementarității acestor sisteme. Sistemele expert și-au pus în mod semnificativ amprenta asupra evoluției sistemelor de asistare a deciziilor, determinându-i pe cercetători să construiască sisteme suport care să nu permită doar accesul la informațiile utile, simularea consecințelor anumitor ipoteze prin utilizarea modelelor existente în baza de modele sau prezentarea rezultatelor într-o formă adaptată necesităților utilizatorului ci, în plus, să fie capabile să ofere o soluție pe care să o justifice. În felul acesta au apărut sistemele expert de asistare a deciziilor. În această direcție am încercat să realizăm dezvoltarea sistemului de asistare a deciziilor propus de mine în capitolul 5. Facilitatea de a face propuneri de investiții pe baza unor criterii decizionale împinge sistemul de asistare a deciziilor spre domeniul sistemelor expert.

9. Domeniul financiar-bancar ocupă o pondere însemnată în cadrul sistemelor expert aflate în exploatare cu rezultate foarte bune. Băncile folosesc sisteme inteligente în luarea deciziilor de investiții, în evaluarea creditelor bancare, detectarea cărților de credit false, evitarea fraudelor etc. Firmele pot folosi sisteme inteligente în planificarea distribuției și în planificarea vânzărilor.

Peste 40% din băncile franceze dezvoltă sau exploatează sisteme expert, iar printre cele mai utilizate sisteme expert în acest domeniu sunt cele legate de recomandări în plasamentul creditelor ținând seama de o serie de factori cum sunt: profilul clienților, patrimoniul, starea fiscală etc. În SUA un asemenea sistem și-a dovedit utilitatea în depistarea fraudelor și a cererilor suspecte. Ceea ce este foarte important de reținut este că mergând în paralel cu experții umani, soluțiile date de sistemul expert au fost aceleași în proporție de 96,5%.

În prezent există o diversitate de sisteme expert de credit ce funcționează pe piața bancară internațională. Un exemplu în acest sens este reprezentat de Expert Credit System implementat de o instituție financiară americană (IIT). Acest sistem este utilizat în cadrul procesului de analiză a creditului servind analistului de credit ca un sistem de asistare a deciziilor.

Folosirea sistemelor expert câștigă repede teren și în domeniul contabilității. Toate marile firme internaționale de contabilitate folosesc sisteme expert în practică sau au sisteme în dezvoltare. Acestea sunt folosite în general pentru asistare într-o varietate de sarcini, în impozite și taxe, în audit etc.

10. La noi în țară nu există încă o piață a sistemelor expert. Costurile care la prima vedere par mari, fără să se țină cont de avantajele aplicării lor care ar duce la recuperarea într-un scurt timp a investiției, descurajează firmele aflate oricum în plin proces de restructurare.

Neutilizarea sau utilizarea într-o mică măsură a sistemelor expert duce la lipsă de eficiență în activitatea firmelor și în cele din urmă la pierderi. Sistemele din domeniul finanțelor, în special al creditării, nu lipsesc din nici o bancă din țările dezvoltate. La noi, ignorarea mijloacelor științifice de calcul a eficienței afacerilor a făcut să fie creditate multe afaceri care până la urmă s-au dovedit a fi nerentabile. Slaba analiză a pieței, a resurselor firmelor au permis investiții care au dus la faliment firmele care le-au făcut. Considerăm că una din soluțiile care pot stimula implementarea unor astfel de sisteme este mult așteptata lege a leasingului în domeniul software.

11. Așteptările unora față de ceea ce a oferit și oferă în continuare sistemele inteligente s-au dovedit uneori nefondate. Un exemplu concludent pentru eșecul implementării unor astfel de sisteme este proiectul Cyc. Demarat în anul 1984, acest proiect a încercat să asimileze cunoștințele uzuale ale oamenilor, prin introducerea manuală de relații între simboluri, cum ar fi "Păsările au pene" sau "Mamele sunt mai în vârstă decât copiii lor". Inițiatorii acestui proiect și-au propus ca sistemul să ajungă în anul 1994 să poată asimila singur date, procesând cărți și studii. După ce a consumat mai mult de 60 de milioane de dolari (printre investitori numărându-se și Microsoft) și 5 secole-persoane de introducere de date, proiectul nu a reușit nici astăzi ceea ce și-a propus inițial.

Inteligența artificială nu poate fi în nici un caz privită ca un panaceu universal. Nu orice proiect poate fi abordat din această prismă și există multe cazuri în care proiecte ambițioase au eșuat. În concluzie conștientizarea limitelor acestui domeniu este obligatorie înainte de a începe orice demers în această privință. Succesul acestor aplicații depinde în mare măsură de corectitudinea informațiilor de care se dispune dar și de disponibilitatea factorilor de decizie de a accepta și de a acționa având la bază concluziile acestor sisteme inteligente.

12. Domeniul Inteligenței Artificiale și sistemele expert în particular rămân unele din cele mai mari promisiuni ale domeniului tehnologiei informației. Mulți specialiști sunt implicați în crearea unor aplicații care utilizează tehnologii ce țin de inteligența artificială. În ultimii ani progresele în domeniu au permis apariția unor aplicații foarte evolute, apărând speculații despre posibilitatea realizării unui sistem care să se apropie mult de limbajul natural.

Datorită dinamicii acestui sector și datorită sumelor mari implicate în special în domeniul investițional, considerăm că orice firmă trebuie să țină pasul cu progresele informaticii în general și cu instrumentele oferite de inteligența artificială în special. Dacă economia românească nu se va adapta

cerințelor pieței și unei concurențe ce dispune de un avans considerabil, nu se va înregistra un reviriment economic real și într-un timp rezonabil.

8.2. DIRECȚII DE VIITOR ALE CERCETĂRII

Studiile efectuate în această teză de doctorat mi-au permis deducerea următoarelor direcții posibile și necesare de cercetare:

1) Integrarea sistemului expert cu alte sisteme

Într-o economie caracterizată de o complexitate și concurență tot mai accentuate, câștigători vor fi cei care au acces la cât mai multe informații și adoptă decizii în timp real. Administrarea eficientă a companiilor impune găsirea de răspunsuri rapide și corecte la multitudinea de probleme care apar la gestionarea resurselor. Soluția o reprezintă sistemele informatice integrate cu menirea de a optimiza procesul decizional.

Se știe că sistemele expert au fost dezvoltate în general în limbaje de programare sau medii neconvenționale și uneori rulează pe diverse echipamente nestandard. Este necesar un efort deosebit pentru ca tehnologia sistemelor expert să poată fi integrată cu tehnologiile altor tipuri de aplicații. Marea problemă nu este atât în privința limbajelor și a echipamentelor, ci mai ales în ceea ce privește conceptele și abordările. Cercetătorii vor trebui să acorde o importanță sporită acestei problematice, pentru a permite dezvoltarea în viitor a unor sisteme complexe care să aibă în componența lor și sisteme inteligente.

În prezent, rareori aplicații “stand-alone” oferă beneficiile așteptate. Se preconizează creșterea ponderii pe care o vor avea sistemele hibride, ce încorporează componente de statistică avansată, euristică, logică fuzzy, rețele neuronale, interfețe spre limbaje naturale etc. Competiția de pe piețele financiare încurajează aceste experimente ducând la dezvoltarea unor sisteme sau cel puțin prototipuri deosebit de sofisticate. Avantajul de care poate beneficia astfel macroeconomia este creșterea eficienței investițiilor. De aceea, organizațiile trebuie să privească foarte atent problematica integrării diverselor tipuri de sisteme.

2) Integrarea bazelor de cunoștințe cu bazele de date

În general, sistemele expert nu interacționează cu sisteme de tip Data Base Management System (DBMS). Această problemă a preocupat mult comunitatea de cercetători. Rezultatele acestor cercetări au fost transferate în unele DBMS-uri comerciale. Astfel, atât INGRES cât și Sysbase au integrat sisteme sofisticate de reguli în motoarele DBMS-urilor lor relaționale.

În practică se întâlnesc nenumărate cazuri în care bazei de cunoștințe trebuie să i se asocieze o bază de date, în general relațională. În aceste cazuri se pune problema consistenței dintre cele două surse de date. Este deseori dificil de a aduce baza de date în starea în care să reflecte modificările ce pot apărea în baza de cunoștințe. O interfață care să permită o interacțiune facilă către sistemul expert și baza de date poate îmbunătăți dar nu rezolvă în totalitate problema consistenței între cele două surse de date. Soluția reprezintă o abordare integrată în care un singur sistem să administreze ambele tipuri de obiecte.

3) Achiziția automată a cunoștințelor

Se cunoaște că actualizarea continuă a bazei de cunoștințe este mai costisitoare decât dezvoltarea sistemului expert în sine. Necesitatea întreținerii și a lărgirii bazei de cunoștințe devine o necesitate din ce în ce mai stringentă. Se impune în acest caz crearea unui mecanism de achiziție automată de cunoștințe.

Achiziția automată de cunoștințe este îngreunată de lipsa unor standarde de raportare financiară on-line. În prezent nu există posibilitatea de achiziție automată de date financiare de pe diversele site-uri decât utilizând proceduri specifice pentru fiecare caz în parte. O posibilă soluție la această problemă este utilizarea XBRL (Extended Business Reporting Language) care este un standard XML (Extensible Markup Language) creat special pentru a rezolva problemele de raportare în domeniul afacerilor. Cercetările în această direcție se pot solda cu noi perspective pentru sistemele bazate pe cunoștințe.

4) Mineritul datelor și analiza multidimensională pentru extragerea de informații calitative dintr-un depozit de date

Evoluția tehnologiei informației a ajuns să ne pună la dispoziție o imensă rețea de calculatoare interconectate la scară mondială și odată cu aceasta accesul la resurse documentare de mare valoare. Mai mult decât atât, odată cu trecerea timpului, volumul datelor este într-o creștere exponențială și ca urmare este nevoie de instrumente tot mai puternice de căutare a informațiilor în aceste imense depozite de date ("Data Warehousing"). Considerăm necesară crearea unor centre de colectare a informațiilor organizate după cerințe specifice care să culeagă datele din diverse domenii și să le pună apoi la dispoziția utilizatorilor interesați. La rândul lor aceste centre de informare și documentare vor deveni depozite sau baze de date documentare dar mult mai bine organizate și pregătite să răspundă prompt solicitărilor venite din partea utilizatorilor.

În acest sens, informația dintr-un depozit de date se poate valorifica prin două modalități: "mineritul" în date (Data Mining) și analiza multidimensională (prin intermediul unor instrumente de tip On Line Analytical Processing -OLAP).

Tehnicile de data mining permit identificarea unor corelații (patterns) în cadrul unor seturi de date, pe baza cărora se pot crea modele predictive care să anticipeze comportamente sau anumite evenimente bazate pe trenduri găsite în depozitele de date. Instrumentele data-mining utilizează metode statistice și bazate pe reguli pentru identificarea anumitor tipare și generarea de modele de evoluție. Se utilizează tehnici complexe, de diverse facturi (inteligență artificială, baze de date, logică și statistică matematică, etc). Se spune că mineritul în date răspunde la întrebări pe care analistul nici măcar nu și le pune.

Alături de acestea stau instrumentele OLAP ce realizează prelucrarea analitică în timp real a informațiilor dintr-un sistem. Rezultatele oferite de acestea nu mai necesită nici o prelucrare ulterioară, putând fi folosite direct în procesul decizional. Ele permit ca din datele și informațiile colectate să se obțină cunoștințe, totul în scopul creșterii profiturilor, îmbunătățirii competitivității sau pur și simplu doar pentru a supraviețui pe piață.

Volumul imens de date care se acumulează precum și dezvoltarea exponențială a puterii de calcul a sistemelor hardware (care rămâne în mare parte neutilizată) necesită utilizarea unor instrumente inteligente în vederea extragerii cunoașterii. Depozitele de date vor trebui transformate în depozite de cunoștințe pe baza cărora managerii să ia decizii inteligente. Transformarea depozitelor de date în depozite de cunoștințe este o provocare la care comunitatea cercetătorilor va răspunde cu siguranță.

Toate aceste problematice vor constitui subiectul unor cercetări ulterioare.

BIBLIOGRAFIE

1. [Airinei93] Airinei D. –Sistemele expert în activitatea financiar-bancară, Tribuna Economică, Nr. 11, 1993, Nr. 12, 1993.
2. [Andone93] Andone I. -Inteligența artificială și sisteme expert în contabilitate, Ed. Moldova, Iași, 1993.
3. [Andone94] Andone I. –Sisteme expert. Principii și dezvoltarea aplicațiilor de gestiune, vol. I, Editura Polirom, Iași, 1994.
4. [Andone95] Andone I. –Sisteme expert. Principii și dezvoltarea aplicațiilor de gestiune, vol. II, Editura Polirom, Iași, 1995.
5. [Andone97] Andone I., Țugui A. –Baze de date inteligente în managementul firmei, Editura Dosoftei, Iași, 1997.
6. [Andone99] Andone I., Țugui A. –Sisteme inteligente în management, contabilitate, finanțe-bănci, Editura Economică, București, 1999.
7. [Andone01] Andone I., Dologite D.G., Mockler R., Țugui A. –Dezvoltarea sistemelor inteligente în economie, Editura Economică, București, 2001.
8. [Andone03] Andone I –Metodologiile și mediile de dezvoltare a sistemelor inteligente. Experiența europeană –Simpozion Internațional “Specializare, Dezvoltare și Integrare”, Cluj-Napoca, 2003.
9. [Andrew92] Andrew A. -Investment, Graham&Trotman, Londra, 1992.
10. [Andrew93] Andrew A., Bloomfield D., Booth P., England P. -Investment Mathematics and Statistics, Graham&Trotman, Londra, 1993.
11. [Angas70] Angas L.L.P. -Placements et speculations en bourse, Petit bibliotheque, Ed. Payot, Paris, 1970, 16.
12. [Anghelache92] Anghelache G., Dardac N., Stancu I. -Piețe de capital și burse de valori, Soc. Adevărul S.A., București, 1992.
13. [Anghelache00] Anghelache G. –Bursa și piața extrabursieră, Editura Economică, București, 2000.
14. [Barr86] Barr A., Feigenbaum E.A., -The Handbook of Artificial Intelligence, volumes 1-4, Addison-Wesley, Reading, MA, 1986.
15. [Bădescu98] Bădescu A. –Inteligență artificială și sisteme expert, Editura Era, București, 1998.
16. [Benchimol93] Benchimol G., Levine P., Pomerol J.C. –Sisteme expert în întreprindere, Editura Tehnică, București, 1993.
17. [Boden87] Boden M.A. -Artificial Intelligence and Natural Man, MIT Press, 1987.
18. [Bojan01] Bojan I. B. -Sisteme expert pentru evaluarea întreprinderilor, Editura Dual Tech, 2001
19. [Bologa95a], **Bologa C.**, Danciu D. -Suport informatic pentru managementul certificatelor de proprietate, Concursul "InfoEc XXI-Aplicații ale informaticii în economie" organizat de INFOTIN Craiova 6-9 aprilie 1995 –lucrare premiată cu premiul I
20. [Bologa95b] **Bologa C.** Danciu D. -Suport informatic pentru programul de privatizare în masă, "Tempus Joint European Network Workshop" Cluj-Napoca, 27-29 noiembrie 1995
21. [Bologa96a] **Bologa C.**, Țiclea A. -Suport informatic pentru evidența tranzacțiilor pe piața OTC (Over the Counter), Sesiunea de Comunicări Științifice a Universității "Aurel Vlaicu" 16-17 mai 1996, Arad
22. [Bologa96b] **Bologa C.** -Suport informatic pentru asistarea deciziilor în tranzacțiile cu valori mobiliare, Sesiunea Științifică Jubiliară -Procesul tranziției la economia de piață –abordare teoretică și practică” 24-25 mai 1996, Cluj-Napoca
23. [Bologa96c] **Bologa C.** -Sistem informatic pentru gestiunea portofoliului de valori mobiliare, Sesiunea Științifică a Studenților, Craiova, octombrie 1996
24. [Bologa97a] **Bologa C.** -Sistem expert în domeniul analizei bursiere, Sesiunea Științifică a Universității Dimitrie Cantemir, Târgu-Mureș, 30 mai 1997
25. [Bologa97b] **Bologa C.**, Tomescu G. -Back-office pentru Societățile de Valori Mobiliare, Sesiunea de Comunicări Științifice “Economia României la orizontul anilor 2000”, Cluj-Napoca, 14-15 noiembrie 1997

26. [Bologa98] **Bologa C.** –Sisteme suport de decizie în domeniul investițiilor în valori mobiliare – Lucrare de dizertație, Cluj-Napoca, 1998.
27. [Bologa99] **Bologa C., Nițchi S.** -Utilizarea Inteligenței Artificiale în domeniul analizei bursiere, Sesiunea Științifică a Universității Bogdan Vodă, « Drept, Economie, Societate Civilă, Educație – Perspective pentru mileniul III », Baia Mare 21-22 Mai 1999
28. [Bologa00] **Bologa C.** -Asupra unui prototip de generator de sisteme expert, Sesiunea Jubiliară de Comunicări Științifice « Creștere economică, dezvoltare, progres » Cluj-Napoca 24-25 Noiembrie 2000
29. [Bologa01a] **Bologa C.** -Tehnici financiar bancare inteligente, Sesiunea Științifică a Universității Petru Maior, Tg. Mureș, 23-24 Noiembrie 2001
30. [Bologa01b] **Bologa C.** -Detectarea fraudelor pe piața bursieră și a asigurărilor, Sesiunea Științifică a Universității Petru Maior, Tg. Mureș, 23-24 Noiembrie 2001
31. [Bologa01c] **Bologa C.** –Analiza indicatorilor statistici pentru acțiuni și obligațiuni –Referat în cadrul stagiului de pregătire a doctoratului, Cluj-Napoca, 2001
32. [Bologa02a] **Bologa C.** –Sisteme expert ; Posibilități de utilizare în domeniul bursei -Referat în cadrul stagiului de pregătire a doctoratului, Cluj-Napoca, 2002
33. [Bologa02b] **Bologa C.** –Elaborarea unui sistem expert în domeniul analizei bursiere -Referat în cadrul stagiului de pregătire a doctoratului, Cluj-Napoca, 2002
34. [Bologa02c] **Bologa C.** –Analiza și evaluarea unui sistem informatic managerial, Editura Risoprint, 2002, coordonator : Goron S.
35. [Bologa03] **Bologa C.** –Sisteme expert în domeniul investițional, Simpozionul Internațional « Specializare, Dezvoltare și Integrare », Cluj-Napoca, 14-15 Noiembrie 2003
36. [Bonsi96] Bonsi Claudia -În așteptarea indicelui bursier și Piața OTC -iarmarocul acțiunilor - Revista Capital nr. 4, 1996.
37. [Booch99] Booch, G., Rumbaugh, J. and Jacobson, I. -The Unified Modelling Language User Guide, Addison Wesley, Reading, Massachusetts, 1999.
38. [Brealey91] Brealey R.A., Myers S.C. -Principles of Corporate Finance -Fourth Edition, McGraw-Hill Inc., 1991.
39. [Brenner98] Brenner W., Zarnekow R., Wittig H. –Intelligent Software Agents, Foundation and Applications, 1998.
40. [Buchanan83] Buchanan B.G. –Constructing an Expert System, Eds. Hayes-Roth, Watermann and Lenat, 1983.
41. [Carlin00] Carlin B.P., Louis T.A. –Bayes and Empirical Bayes Methods for Data Analysis – Second Edition Chapman & Hall, CRC, Boca Raton, 2000.
42. [Cârstoiu94] Cârstoiu D.I. –Sisteme expert, Editura All, București, 1984.
43. [Chabris88] Chabris C.F. -A Primer of Artificial Intelligence, Kogan Page, 1988.
44. [Ciobanu97] –Bursele de valori și tranzacțiile la bursă, Editura Economică, București, 1997.
45. [Clocksin81] Clocksin W.F., Mellish C.C. –Programming in Prolog, Springer-Verlag, New York, 1981.
46. [Cockburn01] Alistair Cockburn –Agile Software Development, Addison Wesley, 2001
47. [Cluj96] XXX -Conferința internațională “Perspectivele bursei de valori în țările Europei Centrale și de Est”, Cluj, 22-23 martie 1996.
48. [Collin88] Collin P. -Dictionary of Computing, First Edition, Teddington, Peter Collin Publishing, 1988.
49. [Corduneanu95] –Piețe financiare și operațiuni bursiere, Editura Mirton, Timișoara, 1995.
50. [Coroescu02] Coroescu T –Sisteme informatice pentru management, Editura Lumina Lex, București, 2002.
51. [Davidescu93] Davidescu D.N. –Utilizarea sistemelor expert în domeniul financiar-contabil, Editura Didactică și Pedagogică, București, 1993.
52. [Davidescu98] Davidescu D. N. și alții –Sisteme informatice financiar-bancare, Editura All Beck, București, 1998
53. [Davis57] Davis M. -A Computer Program for Pressburger's Procedure. Summer Institute for Symbolic Logic, 1957.

54. [Davis82] Davis R., Lenat D.B. –Knowledge-based Systems in Artificial Intelligence, McGraw-Hill, New York, 1982.
55. [Defosse90] –Defosse G, Balley P –La bourse des valeurs, Presses Universitaires de France, Paris, 1990
56. [Dodescu96] Dodescu G., Odăgescu I., Năstase P., Scheianu S. –Simularea sistemelor, Editura Militară, București, 1996.
57. [Dodescu87] Dodescu G., Apostol C., Roșca I., Odăgescu I. –Informatica, Editura Științifică și Enciclopedică, București, 1987.
58. [Dragomirescu96] Dragomirescu H. –Strategii de informatizare –în Strategii manageriale de firmă (coordonator Nicolescu O.), Editura Economică, București, 1996.
59. [Drăgoescu95] Drăgoescu Elena, Drăgoescu A. -Piețe financiare primare și secundare și operațiuni de bursă, Editura Mesagerul, Cluj-Napoca, 1995.
60. [Drăgănescu80] Drăgănescu M -A doua revoluție industrială: microelectronica, automatica, electronica - factori determinanți, Editura Tehnică, București, 1980.
61. [Dubois88] Dubois D, Prade H –Possibility Theory –Approach to Computerized Processing of Uncertainty, Plenum Press, New York, 1988.
62. [Elcock67]Elcock E.W., Murray A.M. -Experiments with a Learning Component in a GO-MOKU Playing Program. In: Collins, N.L. & Michie, D. (Ed.), Machine Intelligence 1, Oliver and Boyd, Edinburgh, 1967.
63. [Feigenbaum83] Feigenbaum E.A., McCorduck P. –The 5th Generation, Addison-Wesley, Reading, Massachusetts, 1983.
64. [Filip00] Filip F.G., Filip C.I. –Decision Support Systems: from Job Aids to Intelligent Assistants. Preprints MIM 2000, IFAC Symposium on Manufacturing, Modelling and Control, July 12-14, Patras, 2000.
65. [Filip01] Filip F.G. –Sisteme de asistare inteligentă a activității manageriale, în Sistemul informațional managerial al organizației –coordonator Nicolescu O., Editura Economică, București, 2001.
66. [Filip04] Filip F.G. –Sisteme suport pentru decizii, Editura Tehnică, București, 2004.
67. [Fischer91] Fischer D.E., Jordan R.J. -Security Analysis and Portfolio Management, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
68. [Fodi99] Fodi A., Constantinescu L., Postăvaru N –Introducere în piața de capital și bursa de valori, Editura Matrix Rom, București, 1999.
69. [Freedmann93] Fredmann R. –AI on Wall Street –The Encyclopedia of Computer Technology, Volume 28, Marcel Dekker Inc., New York, 1993.
70. [Frențiu94] Frențiu M., Pârv B. -Elaborarea programelor: Metode și tehnici moderne, Editura Promedia, 1994.
71. [Gallois97] –Gallois D –Bursa. Origine și evoluție, Editura Teora, București, 1997
72. [Georgescu85] Georgescu I. –Elemente de inteligență artificială, Editura Academiei, București, 1985.
73. [Georgescu95] Georgescu I. -Randamentul investițiilor de capital -Revista Capital nr. 18, 1995.
74. [Ghilic97] Ghilic-Micu B. -Bursa de Valori, Editura Economică, 1997.
75. [Ghișoiu01] Ghișoiu N, Goron S, Tarca N –Baze de date și programare, Editura Risoprint 2001
76. [Ghișoiu04] Ghișoiu N., Nițchi S, Nițchi A.R., Tomai N. –Baze de date și programare, Editura Risoprint, 2004.
77. [Giarratano93] Giarratano J., Riley G. –Expert Systems. Principles and Programming, Designing Expert Systems using CLIPS, PWS Pub, Boston, 1993
78. [Gîrjoabă00] –Gîrjoabă D –Analiza fundamentală versus analiza tehnică pe piața românească, Piața financiară nr. 11, 2000
79. [Gobesz99] Gobesz Z. –Contribuții la dezvoltarea sistemelor expert în domeniul construcțiilor – Teză de doctorat, U.T. Cluj-Napoca, 1999
80. [Goron99] Goron S. –Proiectarea orientată obiect a produselor program, Editura Risoprint, Cluj-Napoca, 1999.
81. [Goron00]Goron S., Joldeș R., Ileană I. –Inițiere în inteligența artificială, Editura Risoprint, Cluj-Napoca, 2000.

82. [Goron02] Goron S., **Bologa C.** -Proiectarea și caracterizarea sistemelor informatice manageriale. Metodologii orientate obiect, Editura Mediamira, Cluj-Napoca, 2002
83. [Goron03] Goron S., **Bologa C.**, Buchmann R. –Proiectarea și caracterizarea sistemelor informatice manageriale, Editura Risoprint, Cluj-Napoca, 2003.
84. [Gorry71] Gorry, G.A., M.S. Scott-Norton -A framework for management information systems. Sloan Management Review, 1971
85. [Goldberg83] Goldberg A., Robson D. –Smalltalk-80 –The Language and its Implementation, Addison-Wesley, 1983.
86. [Goonatilake95] Goonatilake S. -Intelligent Systems for Finances and Business: An Overview, John Wiley&Sons, 1995.
87. [Goron98] Goron S. –Elemente de ingineria produselor program, Editura Risoprint, 1998.
88. [Graham88] Graham I. –Expert Systems, Knowledge, Uncertainty and Decisions, Chapman and Hall, 1988.
89. [Griffith87] Griffith M., Palissier C. -Algorithmic Methods for Artificial Intelligence. Kogan Page, 1987.
90. [Hanart90] –Hannart G –Prevision des cours de bourse, Sefi, Montreal, 1990
91. [Hayes83] Hayes-Roth F., Watterman D.A. Lenat D.B. –Building Expert Systems, Addison-Wesley, Reading, 1983.
92. [Hußmann97] H. Hußmann: Formal Foundations for Software Engineering Methods. LNCS 1322, Springer-Verlag, Berlin, Germany, 1997
93. [Hunt75] Hunt E.B., -Artificial Intelligence, New York, Academic Press, 1975.
94. [Intrados95] XXX -Manual de pregătire a brokerilor -elaborat de ANSVM-Grupul Intrados, 1995.
95. [Ivan99] Ivan I, Nosca Gh., Parlog O., Tcaciuc S. -Calitatea datelor, Editura Infosec, București, 1999
96. [Ivan01] Ivan I., Teodorescu L. –Managementul calității software, Ed. Infosec, București, 2001
97. [Ivan04] Ivan I., Boja C.– Metode statistice în analiza software, Editura ASE, București, 2004
98. [Jeffery92] Jeffery K. -Expert Database Systems, Academic Press Inc., San Diego, 1992.
99. [Jojo94] Jojo L., O'Keefe R.M -Experiences with an expert system prototyping methodology” în Expert Systems: The International Journal of Knowledge Engineering, 1994.
100. [Kaufmann87] Kauffman A. –Nouvelles logiques pour l’intelligence artificielle, Hermes, Paris, 1987.
101. [Keefe93] O'Keefe R.M., O'Leary D.E. -A review and survey of expert system verification and validation, Artificial Intelligence Review 7:1, 1993.
102. [Keen76] Keen P.G. –Interactive computer systems for managers: a modest proposal. Sloan Management Review, Fall, 1976
103. [Kent61] Kent A. -Information Retrieval and Mechanical Translation. Vol. 1-2, Interscience Publishers, 1961
104. [Klein90] Klein M., Methlie L.B. –Expert Systems. A Decision Support Approach, Addison-Wesley Publishing Co., 1990.
105. [Knight90] Knight K., Rich E. –Artificial Intelligence - McGraw-Hill, 1990
106. [Kovacs90] Kovacs A. -Les strategies de la nouvelle bourse, Les Editions d’organisations, Paris, 1990.
107. [Levine89] Levine P., Pomerol J. –Sistemes interactifs d’aide a la decision et sytemes experts, Hermes, Paris, 1989.
108. [Liebowitz98] Liebowitz J. –The Handbook of Applied Expert Systems, CRC Press LLC, Boca Raton, 1998.
109. [Lițoiu98] Lițoiu V, Bădică A., Tărbujaru V. –Sisteme expert financiar-bancare –curs, Reprografia Universității din Craiova, 1998
110. [Malița87] Malița M. –Bazele inteligenței artificiale, Editura Tehnică, București, 1987.
111. [Marakas03] Marakas G.M. –Decision Support Systems and Megaputer, Prentice Hall, New Jersey, 2003.
112. [Marinescu98] Marinescu G. -Timpul și deciziile manageriale, Editura Junimea, Iași, 1998.

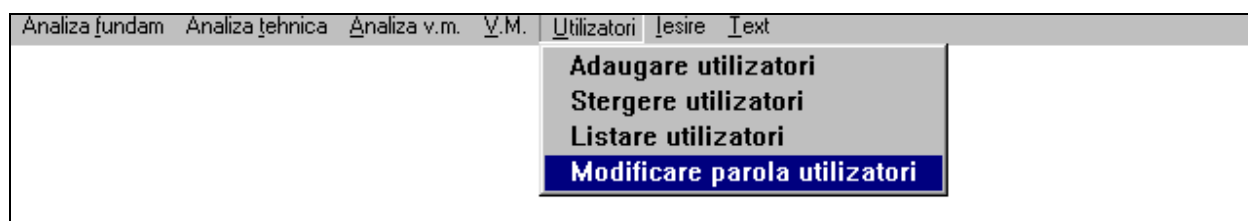
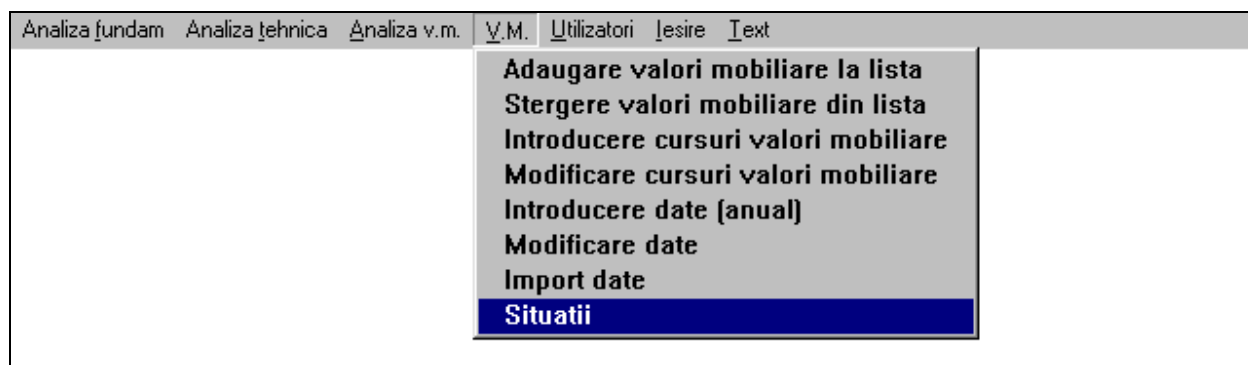
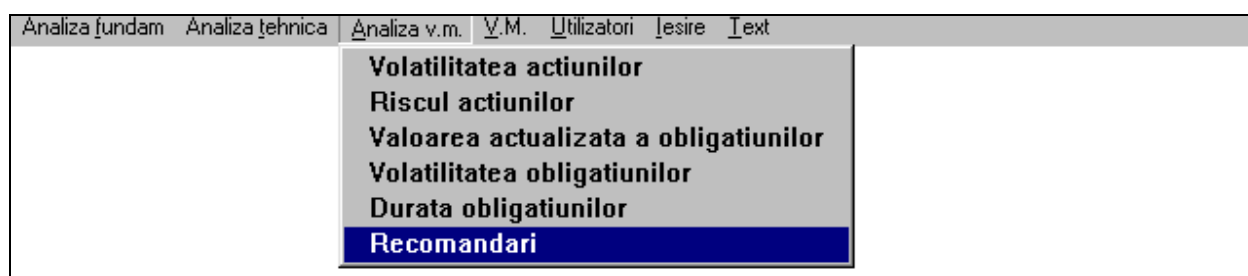
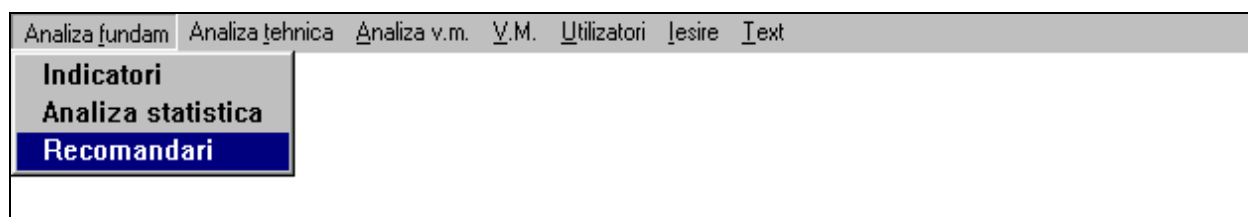
113. [Marshall90] Marshall G. –Advanced Students Guide to Expert Systems, Heinemann Newnes, 1990.
114. [Mărăcine98] Mărăcine V. –Decizii manageriale, Editura Economică, București, 1998.
115. [Meszaros96] Meszaros Judith –Turbo Prolog 2.0 –Ghid de utilizare, Editura Albastră, Cluj-Napoca, 1996.
116. [Michaelsen83] Michaelsen R., Michie D. –Expert Systems in Business, Datamation, 1983.
117. [Michie84] Michie D., Muggleton S., Riese C., Zubrick S. –RuleMaster: A Second Generation Knowledge Engineering Facility. Presented at the First Conference on Artificial Intelligence Applications, Denver, Colorado, 5-7 December 1984.
118. [Millet92] Millet I., Mawhinney C. –Executive Information Systems, Information & Management, 1992.
119. [Minsky74] Minsky M. -A Framework for Representing Knowledge, MIT AI Laboratory Memo 306, June 1974.
120. [Mintzberg90] Mintzberg H. –Planning on the Left Side and Managing on the Right. In Creative Management (ed. J. Henry) SAGE Publications, London, 1990.
121. [Morton71] Morton S –Management Decision Systems: computer based support for decision making; Harvard University Press, Cambridge, 1971.
122. [Newell57] Newell A., Shaw J.C., Simon H. -Empirical Explorations with the Logical Theory Machine. Proceedings of WJCC, 1957.
123. [Nițchi94] Nițchi S. -Esential în dBase, Foxpro și Clipper, Editura Promedia, Cluj-Napoca, 1994.
124. [Nițchi97] Nițchi S., Nițchi R. -Data mining, o nouă eră în informatică, BYTE, 3, 1997, 117-121.
125. [Nițchi03a] Nițchi S –Curs de Inteligență Artificială, secția Informatică Economică.
126. [Nițchi03b] Nițchi R.A., Nițchi S., **Bologa C.**, și alții –Culegere de exerciții și probleme la disciplina “Baze de date și programarea calculatoarelor”, Cluj-Napoca, 2003
127. [Nof81] Nof S.Y. - Theory and practice in decision support for manufacturing control; Data Base Management: Theory and Applications, 1981
128. [Nuham95] Nuham V. -Cumpărătorul unei acțiuni devine investitor și Ghidul practic al investitorului la bursă -Revista Capital nr. 27-29, 1995.
129. [Obreja00] Obreja C., Anghelache G. –Piețe de capital și tranzacții bursiere, Editura ASE București, 2000
130. [Olteanu03] Olteanu A., Olteanu M.A. –Piețe de capital, Editura Dareco, București, 2003.
131. [Oprean94] Oprean D., Racovițan D.M., Oprean V. -Informatică de gestiune și managerială, Editura Eurounion, Oradea, 1994, 173-181.
132. [Oprean01] Oprean D, Abdel Rahman Mohamad –IT&C Informare și comunicare, management și informatizare; Proiectarea de tehnologii informațional-decizionale, Editura Risoprint, Cluj-Napoca 2001
133. [Parsaye88] Parsaye K., Chignell M. –Expert Systems for Experts, John Wiley, New York, 1988.
134. [Parsaye89] Parsaye K. -Intelligent Databases, John Wiley, 1989.
135. [Paterson82] Paterson A. –ACLS User Manual, Intelligent Terminals Ltd., 15 Canal St. Oxford, OX2 6BH, UK, 1982.
136. [Pop95] Pop Cornelia -Valoarea de piață a firmei în oglinda indicatorilor bursieri -Revista Tribuna Economică 1995, 22 (24), 24 (24) .
137. [Pop96] Pop Cornelia -Tipuri de acțiuni agreate de piețele financiare -Revista Tribuna Economică 1996, 6 (26), 7 (23), 8 (27), 9 (30).
138. [Popa93] Popa I -Bursa, vol I, Colecția Bursa, București, 1993.
139. [Racovițan96] –Racovițan D, Nitchi S și colectiv–Bazele prelucrării informațiilor și tehnologie informațională, Editura Intelcredo, Deva, 1996.
140. [Racovițan00] Racovițan D., Rusu L., **Bologa C.** și alții -Raport anual de cercetare -”Instruirea cu metode clasice și ale inteligenței artificiale în domeniul informaticii economice și de afaceri” – Tema 48 cod 340 CNCSIS contract 3993/2000
141. [Rich83] Rich E –Artificial Intelligence, McGraw-Hill, New York, 1983.
142. [Rusu01] Rusu L., Pop C., **Bologa C.** și alții -Raport anual de cercetare -”Determinarea indicatorilor calității” –tema 65 –cod 424 CNCSIS –2001, contract 31400/7053.

143. [Samuels95] Samuels J.M., Wilkes F.M., Brayshaw R.E. -Management of Company Finance, Chapman&Hall, 1995.
144. [Sambotin97] Sambotin C. -Sisteme expert cu Prolog, Editura Tehnică, București, 1997.
145. [Samuels59] Samuels A. L. -Some Studies in Machine Learning Using the Game of Checkers. IBM Journal of Research and Development, 3, 1959
146. [Schreiber94] Schreiber A. T., Wielinga B. J., ș.a. -CommonKADS: A comprehensive methodology for KBS development, IEEE Expert 9,6, 1994.
147. [Shannon50] Shannon C. E. -Programming a Computer for Playing Chess. Philosophical Magazine, 7, 1950.
148. [Shapiro87] Shapiro S. -Encyclopedia of Artificial Intelligence. Ed. Stuart C. Wiley - Interscience Publication, vol. 1, 1987.
149. [Sheridon80] Sheridan A. -Decision Support Systems. Current Practices and Continuing Challenges. Addison Wesley, Reading, Massachusetts, 1980.
150. [Simon60] Simon, H -The New Science of Management Decision, New-York, Harper & Row, 1960.
151. [Sloman78] Sloman A -The Computer Revolution in Philosophy (Harvester Studies in Cognitive Science), The Harvester Press, Ltd, 1978.
152. [Sprague82] Sprague R.H., Carlson E.D. -Building Effective Decision Support Systems, Prentice Hall, New Jersey, 1982.
153. [Stoica95] Stoica M. -Piețe de capital pentru firmele mici -Revista Capital nr. 27, 1995.
154. [Turban90] Turban E. -Decision Support and Expert Systems, Macmilan Publishing Co., 1990.
155. [Turban01] Turban E. Aronson J.E. -Decision Support Systems and Intelligent Systems, Prentice Hall, 2001.
156. [Turing50] Turing A.M. -Computing Machinery and Intelligence, Oxford University Press, vol. LIX, no.236, 1950.
157. [Turnier89] Turnier J.C.-Gerer un miniportofeuille en bourse, Les Editions d'organisations, Paris, 1989.
158. [Vad85] Văduva I., Baltac V., Florescu V., Florică I., Jitaru M. -Ingineria programării, Editura Academiei RSR, București, 1985.
159. [Warwick91] Warwick K. -Applied Artificial Intelligence, Peter Peregrinus Ltd, London, 1991.
160. [Waterman86] Waterman D.A. -A Guide to Expert Systems, Addison-Wesley, Reading, 1986.
161. [Weizenbaum66] Weizenbaum J.L. -A Computer Program for the Study of Natural Language Communication between Man and Machine. CACM 0, 1, 1966.
162. [Winston92] Winston P.H. -Artificial intelligence (3rd ed.) Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1992
163. [Wooldridge95] Wooldridge M, Jennings N.R. -Agent Theories, Architectures, and Languages: a Survey, in Wooldridge and Jennings Eds., Intelligent Agents, Berlin, Springer-Verlag 1995
164. [Ying96] Ying C.C. -Stock Market Prices and Volumes of Sales, Econometrica, July, 1996.
165. [Zaharie93] Zaharie D., Năstase P. -Sisteme expert de gestiune, Editura Romcart, București, 1993.
166. [Zaharie93b] Zaharie D, Năstase P. -Sisteme expert de gestiune, Revista "Tribuna Economică" nr. 10/1993.
167. [Zaharie99] Zaharie D., Năstase P. și alții -Sisteme expert: Teorie și aplicații, Editura Dual Tech, 1999
168. [Zaharie01] Zaharie D., Albescu F. și alții -Sisteme informatice pentru asistarea deciziei, Editura Dual Tech, 2001
169. [Zimm89] Zimmermann H.J. -Fuzzy Stes, Decision Making and Expert Systems, Kluwer Academic Pub., 1989.
170. [***] A.G. Edwards <http://www.agedwards.com/>
171. [***] Acquire <http://vvv.com/ai/>
172. [***] Acquire <http://www.iit.nrc.ca/fuzzy/files.html>
173. [***] Agentia de presă Mediafax <http://www.mediafax.ro/capital>
174. [***] AI on the Web <http://www.cs.berkeley.edu/~russell/ai.html>
175. [***] AI Repository <http://www.cs.cmu.edu/groups/ai/html/repository.html>

176. [***] American Association for Artificial Intelligence <http://www.aaai.org>
177. [***] Angoss Knowledge Seeker <http://www.angoss.com>
178. [***] Arity Expert Development Package <http://www.arity.com/www.pl/products/ap.htm>
179. [***] Armchair Millionaire <http://www.armchairmillionaire.com>
180. [***] Art Enterprise <http://www.ai-cbr.org/tools/brightware.html>
181. [***] Art Enterprise <http://www.brightware.com>
182. [***] Bear Stearns <http://www.bearstearns.com/>
183. [***] Beiley Software <http://www.beiley.com>
184. [***] Bibliographies on Artificial Intelligence <http://liinwww.ira.uka.de/bibliography/Ai/>
185. [***] BigCharts <http://www.bigcharts.com>
186. [***] Bucharest Stock Exchange <http://www.bvb.ro>
187. [***] Bursa Monetar Financiară și de Mărfuri Sibiu <http://www.bmfms.ro/>
188. [***] Capital Online <http://www.capital.ro>
189. [***] CBS MarketWatch <http://cbs.marketwatch.com/>
190. [***] Certinvest <http://www.certinvest.ro>
191. [***] CLIPS <http://www.siliconvalleyone.com/clips.htm>
192. [***] CMU Artificial Intelligent Repository
193. [***] CNN Financial News <http://www.cnnfn.com/>
194. [***] Comdale <http://mining.ubc.ca/product/>
195. [***] Creative Tools and Techniques CATWOE <http://www.mycoted.com/creativity/techniques/catwoe.php>
196. [***] Cybertrader <http://www.cybertrader.com/cybertrader/>
197. [***] DLjDirect <http://www.dljdirect.com>
198. [***] Easy Reasoner <http://www.haley.com/>
199. [***] ERS <http://www.ers.com>
200. [***] Esignal <http://www.esignal.com>
201. [***] Exsys Developer <http://www.exsys.com/Exdev.html>
202. [***] FinanCenter <http://www.financenter.com/>
203. [***] Fortune.com <http://cgi.pathfinder.com/fortune/>
204. [***] Fund Manager <http://www.beiley.com/fundman/>
205. [***] FuzzyCLIPS <http://ai.iit.nrc.ca/fuzzy/fuzzy.html>
206. [***] Goldman Sachs <http://www.gs.com/>
207. [***] Hoover's Online <http://www.hoovers.com>
208. [***] Instrum. de realizare a s.e. <http://www.aurelian.ro/CSIE/Cyb/IA/CPE-EM%20suport%2011-12.htm>
209. [***] Investor <http://www.investitor.ro>
210. [***] Investment Adviser <http://www.iard.com/Default.asp>
211. [***] Invest-o-rama! <http://www.investorama.com/>
212. [***] Investorguide <http://www.investorguide.com>
213. [***] IVillage Moneylife <http://www.ivillagemoneylife.com/money/>
214. [***] Real Life Investing Guide <http://www.rlig.com>
215. [***] Journal of Artificial Intelligence Research <http://www.jair.org/>
216. [***] Kiplinger.com <http://www.kiplinger.com/>
217. [***] Libra Bank <http://www.librabank.ro/Romania/Statistics/capital.htm>
218. [***] Macaulay Duration <http://www.investopedia.com/terms/m/macaulayduration.asp>
219. [***] MarketMind <http://www.marketmind.com>
220. [***] Merrill Lynch <http://www.ml.com/>
221. [***] MIKE <ftp://hcr1.open.ac.uk>
222. [***] MIKE http://kmi.open.ac.uk/people/marc/mike_text.html
223. [***] Mitsubishi Research Institute <http://www.mri.co.jp/e/index.html>
224. [***] Mobal <ftp://ftp.gmd.de/gmd/mlt/Mobal/>
225. [***] Mobal <http://www-2.cs.cmu.edu/afs/cs/project/ai-repository/ai/areas/learning/systems/mobal/0.html>
226. [***] Money.com <http://www.pathfinder.com/money/>

227. [***] Morgan Stanley Dean Witter <http://www.deanwitter.com/>
228. [***] MSN Money Central <http://www.moneycentral.com/>
229. [***] Natl. Association of Investors Corporation <http://www.better-investing.org>
230. [***] Pagina Pieței de Capital din România <http://www.kmarket.ro>
231. [***] Paine Webber <http://www.painewebber.com/>
232. [***] Platinum Technology Inc. <http://www.platinum.com>
233. [***] Portfolio Builder <http://www.portfoliobuilder.net/>
234. [***] Prudential Securities <http://www.prudential.com/>
235. [***] Quicken.com <http://www.quicken.com>
236. [***] Rasdaq, Official Site, Romanian Securities Market <http://www.rasd.ro>
237. [***] Real time Wise <http://real-timewise.com>
238. [***] Research Institute for Artificial Intelligence <http://www.racai.ro>
239. [***] RT Expert <http://www.tele.ucl.ac.be/ELEC2920/1997/agentsP/complet.html>
240. [***] Salomon Smith Barney <http://www.smithbarney.com/>
241. [***] SmartMoney.com <http://www.smartmoney.com>
242. [***] Smith Barney Group <http://www.smithbarney.com>
243. [***] StockPoint <http://www.stockpoint.com>
244. [***] The British Computer Society's Specialist Group on Expert Systems
<http://www.chrisnaylor.co.uk/definition.html>
245. [***] TURBO PROLOG –Programmer's Guide -Borland International Inc. Scoots Valey CA USA, 1986
246. [***] TURBO PROLOG –Reference Guide - Borland International Inc. Scoots Valey CA USA, 1986
247. [***] U.S. News Online <http://www.usnews.com>
248. [***] Vanguard Homepage <http://www.vanguard.ro>
249. [***] WindExS <http://www-2.cs.cmu.edu/Groups/AI/html/faqs/ai/expert/part1/faq-doc-5.html>
250. [***] Worlddata <http://www.worlddata.com>
251. [***] Yahoo! Finance <http://quote.yahoo.com/>
252. [***] Zack's Investment Research <http://www.zacks.com/>
253. [***] Zagury <http://www.zagury.com>

Anexa 1 -Interfața aplicației “StockAssist”



Anexa 2 -Situațiile de ieșire ale aplicației “StockAssist” –Indicatori statistici

Analiza fundam		Analiza tehnica		Analiza v.m.		V.M.		Utilizatori		Iesire	
<p>Selectati valoarea mobiliara dorita:</p> <div style="border: 1px solid black; padding: 5px; width: 200px;"> <div>SIF3</div> <div>SNP</div> </div>											
<p>Introduceti pretul pentru care doriti calculul indicatorilor: 3000</p> <p>Introduceti anul pentru care doriti calculul indicatorilor: 2003</p>											
Vm								Record: 2/2		Exclusive	
								Ins		Num	
										Caps	

Analiza fundam		Analiza tehnica		Analiza v.m.		V.M.		Utilizatori		Iesire		Text	
UN	-Valoare nominala:	1000	lei										
BUPS	-Valoare contabila:	2234.80	lei										
P	-Pret de piata:	3000	lei										
PN	-Profit net:	1440500000	mii lei										
DT	-Dividend total:	943363074	mii lei										
NA	-Numar actiuni:	37734522958	buc										
CA	-Cifra de afaceri:	71345900000	mii lei										
AN	-Activ net:	84329172638	mii lei										
AT	-Activ total:	118088772638	mii lei										
P/BU	-Raport Pret/Valoare contabila:	1.34											
EPS	-Profit net per actiune:	38.17	lei										
DUPS	-Dividend per actiune:	25.00	lei										
DY	-Rentabilitate curenta:	0.83	%										
PER	-Raport Pret/Profit net per actiune:	78.59											
CAPA	-Cifra de afaceri la o actiune:	1890.73	lei										
PSR	-Raport Pret/Cifra de afaceri la o actiune:	1.59											
CP	-Capitalizare de piata:	113203568874.00	mii lei										
ROA	-Profit net/Activ total:	1.22	%										
ROE	-Profit net/Capitaluri proprii:	1.71	%										
RPD	-Rata de plata a dividendelor:	0.65											
								Ins		Num		Caps	

Anexa 3 -Situațiile de ieșire ale aplicației “StockAssist” –Analiza statistică

Analiza fundam		Analiza tehnica		Analiza v.m.		V.M.		Utilizatori		Iesire		Text					
Urmărirea indicatorilor statistici pentru perioada cuprinsa între 01/01/04 si 31/12/04																	
Valoarea minima:						1410.00											
Valoarea maxima:						3330.00											
Valoarea medie:						2231.52											
Dispersia:						55799053.1646											
Abaterea medie patratica:						7469.8764											
Coeficientul de variatie a lui Pearson:						3.3474											
												Ins		Num		Caps	

Anexa 4 -Situațiile de ieșire ale aplicației “StockAssist” –Recomandări de investiții

Analiza fundam		Analiza tehnica		Analiza v.m.		V.M.		Utilizatori		Iesire							
<p>In analiza fundamentala se folosesc mai multi indicatori. Fiecare poate sa influenteze decizia de vanzare sau de cumparare. Unii indicatori prezinta o importanta mai mare. De aceea e nevoie de o ierarhizare a lor. Fiecaruia din indicatorii de mai jos i se va asocia un coeficient de importanta (0-100):</p>																	
1. PBU=Raportul Valoare de piata / Valoare contabila						15											
2. EPS=Profit net per actiune						45											
3. DY=Rentabilitate curenta						20											
4. PER=Raportul Pret/ Castig						35											
5. PSR=Raportul Pret/ Cifra de afaceri la o actiune						10											
6. RPD=Rata de plata a dividendelor						20											
7. ROA=Raportul Profit net / Activ total						10											
8. ROE=Raportul Profit net / Active nete						10											
												Ins		Num		Caps	

<p>Analiza fundam Analiza tehnică Analiza v.m. V.M. Utilizatori Iesire Text</p>
<p>Evolutia PBU (Raport valoare de piata/ Valoare contabila):</p> <p>Societatea SIF 3 TRANSILVANIA PBU in anul 2003: 0.25 PBU in anul 2002: 0.14 PBU in anul 2001: 0.11</p> <p>Societatea SNP PETROM PBU in anul 2003: 0.57 PBU in anul 2002: 0.22 PBU in anul 2001: 0.46</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SIF 3 TRANSILVANIA</p> <p>Evolutia EPS (Profit net per actiune):</p> <p>Societatea SIF 3 TRANSILVANIA EPS in anul 2003: 645.14 EPS in anul 2002: 444.32 EPS in anul 2001: 348.72</p> <p>Societatea SNP PETROM EPS in anul 2003: 38.17 EPS in anul 2002: 60.48 EPS in anul 2001: 69.29</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SIF 3 TRANSILVANIA</p>
<p>Ins Num Caps</p>

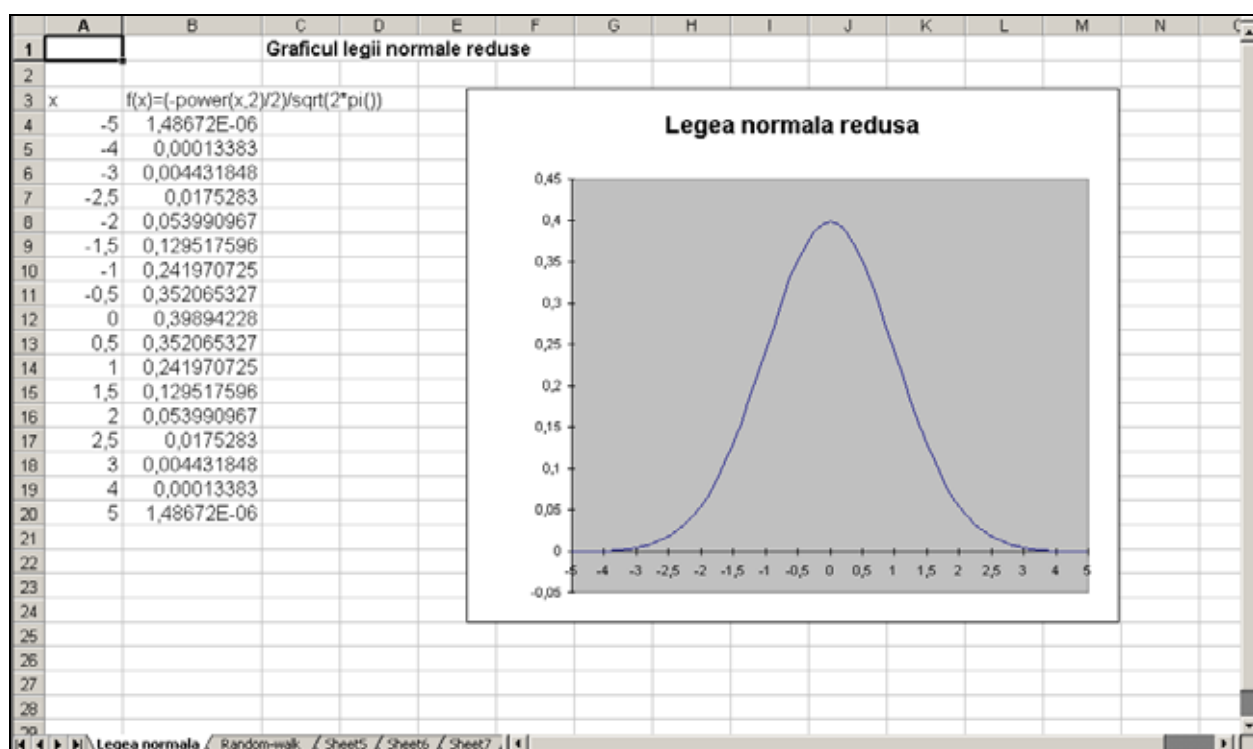
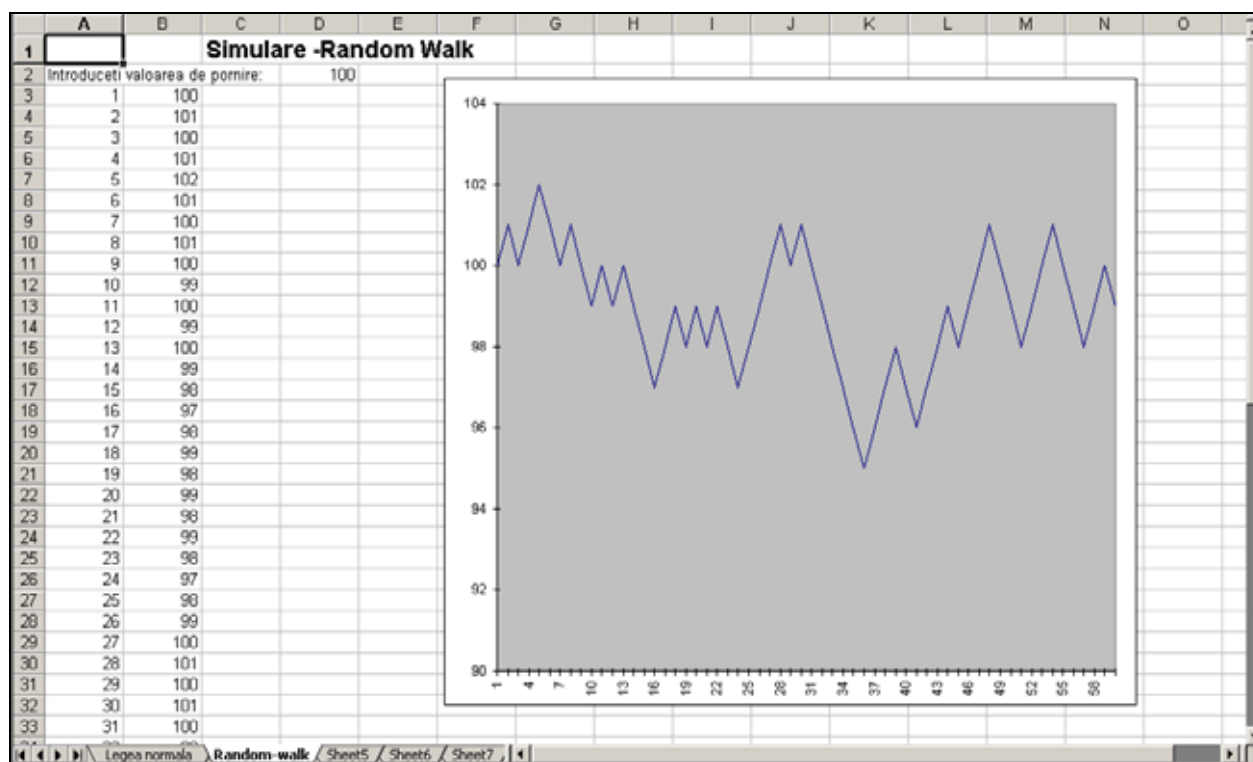
<p>Analiza fundam Analiza tehnică Analiza v.m. V.M. Utilizatori Iesire Text</p>
<p>Evolutia DY (Rentabilitate curenta):</p> <p>Societatea SIF 3 TRANSILVANIA DY in anul 2003: 14.24 DY in anul 2002: 24.11 DY in anul 2001: 38.41</p> <p>Societatea SNP PETROM DY in anul 2003: 1.95 DY in anul 2001: 4.65</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SIF 3 TRANSILVANIA</p> <p>Evolutia PER (Raportul Pret/Profit net):</p> <p>Societatea SIF 3 TRANSILVANIA PER in anul 2003: 4.57 PER in anul 2002: 3.17 PER in anul 2001: 1.98</p> <p>Societatea SNP PETROM PER in anul 2003: 33.53 PER in anul 2002: 6.55 PER in anul 2001: 10.25</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SIF 3 TRANSILVANIA</p> <p>Evolutia PSR (Raportul Pret/Cifra de afaceri la o actiune):</p>
<p>Ins Num Caps</p>

Anexa 4 -Situațiile de ieșire ale aplicației “StockAssist” –Recomandări de investiții

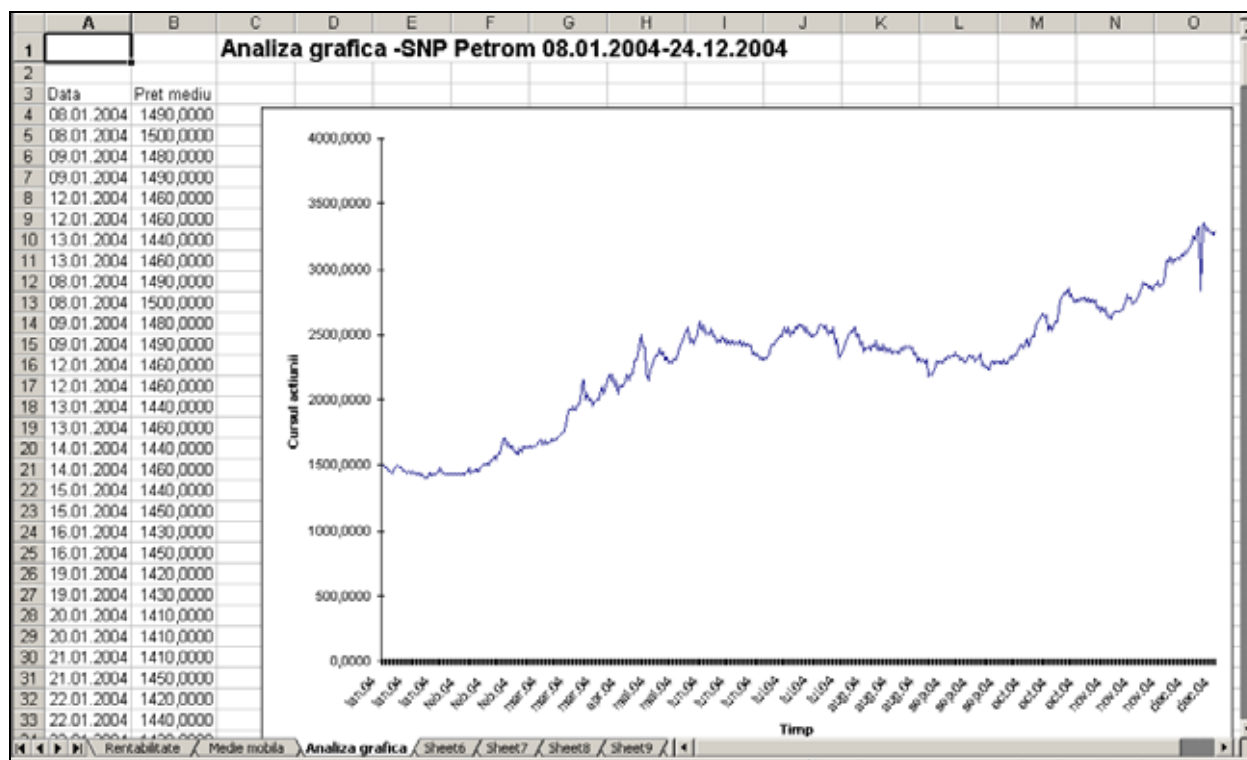
Analiza fundam	Analiza tehnică	Analiza v.m.	V.M.	Utilizatori	Iesire	Text		
<p>Evolutia PSR (Raportul Pret/Cifra de afaceri la o actiune):</p> <p>Societatea SIF 3 TRANSILVANIA PSR in anul 2003: 2.80 PSR in anul 2002: 1.72 PSR in anul 2001: 1.16</p> <p>Societatea SNP PETROM PSR in anul 2003: 0.68 PSR in anul 2002: 0.21 PSR in anul 2001: 0.50</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SNP PETROM</p> <p>Evolutia RPD (Rata de plata a dividendelor):</p> <p>Societatea SIF 3 TRANSILVANIA RPD in anul 2003: 0.65 RPD in anul 2002: 0.77 RPD in anul 2001: 0.76</p> <p>Societatea SNP PETROM RPD in anul 2003: 0.65 RPD in anul 2001: 0.48</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SNP PETROM</p> <p>Evolutia ROA (Raportul Profit net/Activ total):</p>								
						Ins	Num	Caps

Analiza fundam	Analiza tehnică	Analiza v.m.	V.M.	Utilizatori	Iesire	Text		
<p>ROA in anul 2002: 4.34 ROA in anul 2001: 5.23</p> <p>Societatea SNP PETROM ROA in anul 2003: 1.22 ROA in anul 2002: 2.33 ROA in anul 2001: 3.23</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SIF 3 TRANSILVANIA</p> <p>Evolutia ROE (Raportul Profit net/Active nete):</p> <p>Societatea SIF 3 TRANSILVANIA ROE in anul 2003: 5.46 ROE in anul 2002: 4.47 ROE in anul 2001: 5.56</p> <p>Societatea SNP PETROM ROE in anul 2003: 1.71 ROE in anul 2002: 3.43 ROE in anul 2001: 4.48</p> <p>Din punct de vedere al acestui indicator va recomandam investitia in actiuni SIF 3 TRANSILVANIA</p> <p>-----</p> <p>Conform criteriilor formulate de dvs, portofoliul optim consta in:</p> <p>81.82% actiuni SIF 3 TRANSILVANIA 18.18% actiuni SNP PETROM</p>								
						Ins	Num	Caps

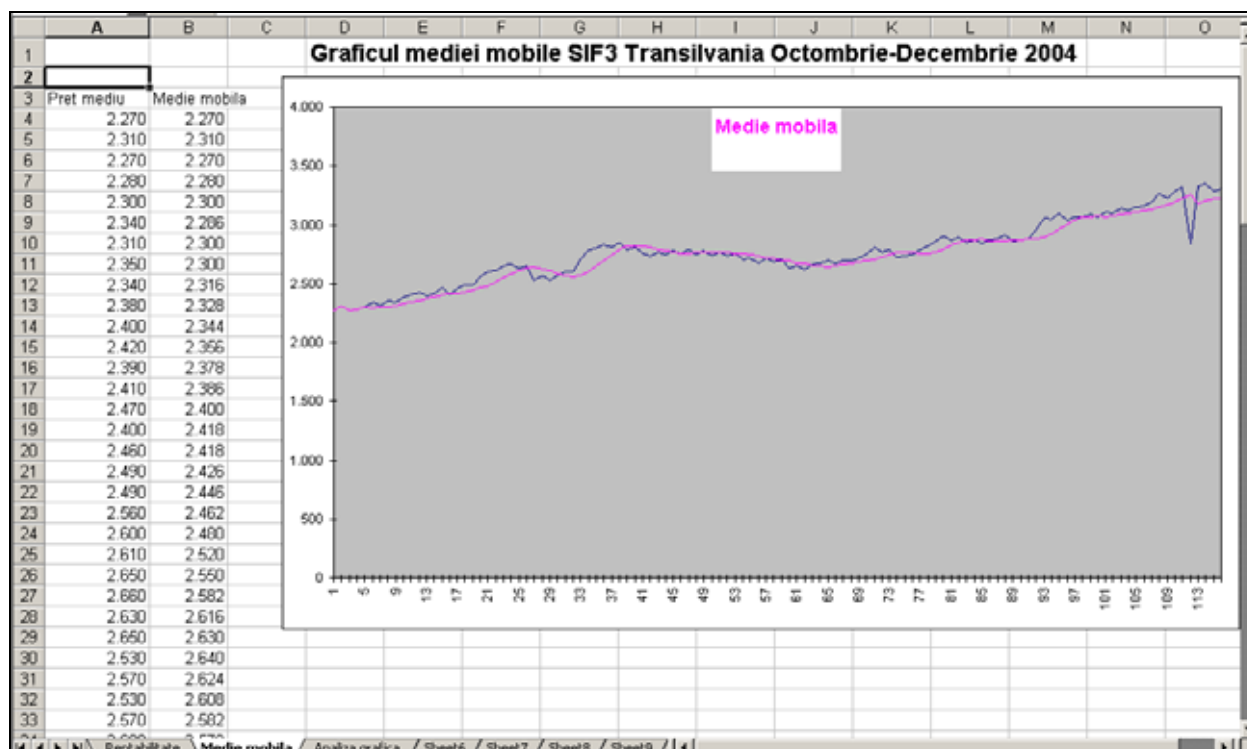
Anexa 5 -Situațiile de ieșire ale aplicației “StockAssist” –Simulare Ranom-Walk; Legea normală redusă



Anexa 6 -Situațiile de ieșire ale aplicației “StockAssist” –Analiza grafică a cursului unei acțiuni



Anexa 7 -Situațiile de ieșire ale aplicației “StockAssist” –Graficul mediei mobile



Anexa 8 -Situatiile de iesire ale aplicatiei “StockAssist” –Valoarea actualizată a obligatiunilor

```

Analiza fundam  Analiza tehnica  Analiza v.m.  V.M.  Utilizatori  Jesire

Introduceti numarul perioadelor de calcul: 3

Perioada          1
Introduceti valoarea: 100
Introduceti factorul de actualizare: 0.11

Perioada          2
Introduceti valoarea: 100
Introduceti factorul de actualizare: 0.15

Perioada          3
Introduceti valoarea: 100
Introduceti factorul de actualizare: 0.09

Valoarea prezenta a acestor fluxuri de numerar este:      240.3

```

Anexa 9 -Situatiile de iesire ale aplicatiei “StockAssist” –Volatilitatea obligatiunilor

```

Analiza fundam  Analiza tehnica  Analiza v.m.  V.M.  Utilizatori  Iesire

Introduceti valoarea nominala a obligatiunii: 1000
Introduceti numarul perioadelor de calcul: 3

Perioada      1
Introduceti rata scadenta din capital: 0
Introduceti dobanda incasata: 100
Introduceti factorul de actualizare: 0.11

Perioada      2
Introduceti rata scadenta din capital: 0
Introduceti dobanda incasata: 100
Introduceti factorul de actualizare: 0.15

Perioada      3
Introduceti rata scadenta din capital: 1000
Introduceti dobanda incasata: 100
Introduceti factorul de actualizare: 0.09

Volatilitatea acestei obligatiuni este:  4.09 %

```

Anexa 10 -Situatiile de iesire ale aplicatiei "StockAssist" –Durata obligatiunilor

Analiza fundam	Analiza tehnica	Analiza v.m.	V.M.	Utilizatori	Iesire
<p>Introduceti valoarea nominala a obligatiunii: 1000 Introduceti numarul perioadelor de calcul: 3</p> <p>Perioada 1 Introduceti rata scadenta din capital: 0 Introduceti dobanda incasata: 100 Introduceti factorul de actualizare: 0.11</p> <p>Perioada 2 Introduceti rata scadenta din capital: 0 Introduceti dobanda incasata: 100 Introduceti factorul de actualizare: 0.15</p> <p>Perioada 3 Introduceti rata scadenta din capital: 1000 Introduceti dobanda incasata: 100 Introduceti factorul de actualizare: 0.09</p> <p>Durata acestei obligatiuni (Duration) este: 2.73</p>					
					<input type="button" value="Ins"/> <input type="button" value="Num"/> <input type="button" value=""/>

Anexa 11 -Situatiile de iesire ale aplicatiei "StockAssist" –Recomandarea unui portofoliu de actiuni

Analiza fundam	Analiza tehnica	Analiza v.m.	V.M.	Utilizatori	Iesire
<p>In functie de coeficientul beta, actiunile se clasifica in: 1)actiuni foarte volatile. Acestea au $\beta > 1$; o crestere (scadere) mai mare de 1% a indicelui general al pietei bursiere provoaca o crestere (scadere) mai mare de 1% a pretului respectivelor actiuni. 2)actiuni putin volatile. Acestea au $\beta < 1$; o crestere (scadere) mai mare de 1% a indicelui general al pietei bursiere provoaca o crestere (scadere) mai mica de 1% a pretului respectivelor actiuni. 3)actiuni cu $\beta = 1$. Evoluaza in acelasi ritm cu piata. Daca doriti o politica agresiva constand in maximizarea profitului va trebui sa va orientati spre actiuni din prima grupa; daca urmariti o politica defensiva, de conservare a capitalului si reducere a riscului, se recomanda investitia in actiuni din a treia grupa.</p> <p>In portofoliul dvs ce pondere(0-100) doriti sa acordati: ategoriei 1): 50 ategoriei 2): 30 Categoriei 3): 20</p>					
					<input type="button" value="Ins"/> <input type="button" value="Num"/> <input type="button" value=""/>

Anexa 11 -Situațiile de ieșire ale aplicației “StockAssist” –Recomandarea unui portofoliu de acțiuni

Analiza fundam	Analiza tehnică	Analiza v.m.	V.M.	Utilizatori	Iesire
----------------	-----------------	--------------	------	-------------	--------

Variabilitatea totala a rentabilitatii unui titlu are 2 componente:
1)risc sistematic. Este determinat de influenta pietei bursiere. El se refera la partea din variabilitatea totala cauzata de factori ce afecteaza pretul tuturor titlurilor.
2)risc specific. E determinat de influenta caracteristicilor specifice ale fiecarui titlu in parte. Riscul specific e partea din riscul total ce e unic pentru o firma sau o industrie.

Riscul se masoara prin coeficientul de corelatie C care masoara gradul de dependenta al actiunii fata de piata. O investitie este cu atat mai riscanta cu cat are un risc specific mai ridicat.

In portofoliul dvs ce pondere(0-100) doriti sa acordati:
 ategoriei 1): 40
 Categoriei 2): 60

	Ins	Num	Caps
--	-----	-----	------

Analiza fundam	Analiza tehnică	Analiza v.m.	V.M.	Utilizatori	Iesire	Text
----------------	-----------------	--------------	------	-------------	--------	------

Rentabilitatea de piata:
 In anul 2003: 8.10
 In anul 2002: 24.11
 In anul 2001: 21.53

Societate: SIF 3 TRANSILVANIA
 Rentab. individuala in anul 2003: 14.24
 Rentab. individuala in anul 2002: 24.11
 Rentab. individuala in anul 2001: 38.41
 Alfa=7.586131 Beta=1.0049
 Coeficientul de corelatie C=0.71114116

Societate: SNP PETROM
 Rentab. individuala in anul 2003: 1.95
 Rentab. individuala in anul 2001: 4.65
 Alfa=0.329000 Beta=0.2006
 Coeficientul de corelatie C=1.00000000

Din punctul de vedere al primului criteriu va recomandam o investitie constand din:
 60.00% de investit in actiuni SIF 3 TRANSILVANIA
 40.00% de investit in actiuni SNP PETROM

Din punctul de vedere al celui de al doilea criteriu va recomandam o investitie constand in:
 40.00% de investit in actiuni SNP PETROM
 60.00% de investit in actiuni SIF 3 TRANSILVANIA

	Ins	Num	Caps
--	-----	-----	------

Anexa 12 -Situatiile de iesire ale aplicatiei "StockAssist" –Volatilitatea unei actiuni

Analiza fundam	Analiza tehnica	Analiza v.m.	V.M.	Utilizatori	Iesire	Text
----------------	-----------------	--------------	------	-------------	--------	------

Societate: SIF 3 TRANSILVANIA
Rentab. individuala in anul 2003: 14.24
Rentab. pietei in anul 2003: 8.10
Rentab. individuala in anul 2002: 24.11
Rentab. pietei in anul 2002: 24.11
Rentab. individuala in anul 2001: 38.41
Rentab. pietei in anul 2001: 21.53

Conform metodei celor mai mici patrate, alfa si beta au urmatoarele valori:
Alfa=7.586131
Beta=1.0049
Ecuatia dreptei de regresie este:
 $ri=7.586131+1.0049rp$
unde ri este rentabilitatea individuala a actiunii alese
iar rp este rentabilitatea generala a pietei.

Alfa semnifica intersectia dreptei de regresie cu axa ordonatelor si reprezinta rentabilitatea actiunii respective atunci cand rentabilitatea pietei este nula.
Beta este coeficientul de regresie sau volatilitatea (numit si coeficient beta).

Ins

Num

Caps

Anexa 13 -Situatiile de iesire ale aplicatiei "StockAssist" –Coeficientul de corelatie

Analiza fundam	Analiza tehnica	Analiza v.m.	V.M.	Utilizatori	Iesire	Text
----------------	-----------------	--------------	------	-------------	--------	------

Societate: SIF 3 TRANSILVANIA
Rentab. individuala in anul 2003: 14.24
Rentab. pietei in anul 2003: 8.10
Rentab. individuala in anul 2002: 24.11
Rentab. pietei in anul 2002: 24.11
Rentab. individuala in anul 2001: 38.41
Rentab. pietei in anul 2001: 21.53

Coeficientul de corelatie este: 0.71114116

Coeficientul de corelatie masoara gradul de dependenta al actiunii selectate fata de piata. Pentru valori pozitive avem o dependenta direct proportionala intre actiunea selectata si piata bursiera iar pentru valori negative relatia de dependenta este invers proportionala. Pentru C=0 nu exista nici o legatura intre rentabilitatea individuala si cea de piata. Pentru C=1, variatia rentabilitatii actiunii este integral si exclusiv datorata variatiei rentabilitatii generale a pietei bursiere (cea mai mare volatilitate si risc specific 0).

Ins

Num

Caps

Anexa 14 -Evoluția cursului unei acțiuni pe o perioadă dată

Analiza fundam Analiza tehnică Analiza v.m. <u>V.M.</u> Utilizatori Iesire Text							
<p>EVOLUTIA PRETULUI ACTIUNILOR SNP de la data de 01/01/04 la data 31/12/04</p>							
Data	Pret mediu	Pret deschid	Pret inchid	Volum tranzac	Valoare tranzactii	Numar tranzac	
08/01/04	1490	1480	1480	49183	73258110	216	
08/01/04	1490	1480	1480	49183	73258110	216	
09/01/04	1480	1480	1480	19644	29096510	82	
09/01/04	1490	1490	1480	614000	916335000	106	
09/01/04	1480	1480	1480	19644	29096510	82	
09/01/04	1490	1490	1480	614000	916335000	106	
12/01/04	1460	1450	1440	12245	17823510	50	
12/01/04	1460	1450	1440	12245	17823510	50	
13/01/04	1440	1440	1450	26026	37512960	95	
13/01/04	1440	1440	1450	26026	37512960	95	
14/01/04	1440	1440	1440	17423	25111830	61	
15/01/04	1440	1440	1420	25645	36928320	100	
15/01/04	1450	1450	1460	634500	920075000	117	
16/01/04	1430	1430	1430	16010	22894300	63	
16/01/04	1450	1450	1450	336500	486445000	85	
19/01/04	1420	1440	1410	13158	18707170	61	
20/01/04	1410	1410	1400	15756	22141830	69	
21/01/04	1410	1390	1440	41695	58852740	154	
22/01/04	1420	1410	1440	21807	31049630	83	
22/01/04	1440	1440	1460	677500	977660000	130	
23/01/04	1430	1430	1440	15698	22417990	73	
26/01/04	1450	1430	1450	20059	29027900	74	

Ins Num Caps

Anexa 15 -Modulele aplicației “StockAssist”

Analiza fundam Analiza tehnică Analiza v.m. <u>V.M.</u> Utilizatori Iesire	
<div>S-a efectuat adaugarea</div> <div> <p>Introduceti simbolul pentru noua valoare mobiliara: <input type="text" value="TLV"/></p> <p>Introduceti descrierea pentru noua valoare mobiliara: <input type="text" value="BANCA TRANSILVANIA"/></p> <p> <input type="button" value="Salvez"/> <input type="button" value="Nu salvez"/> </p> </div>	
Ins Num	

Anexa 15 -Modulele aplicației “StockAssist”

Analiza fundam Analiza tehnică Analiza v.m. V.M. Utilizatori Iesire

S-a efectuat stergerea

Selectati valoarea mobiliara ce va fi stearsa din lista:

Descriere: BANCA TRANSILVANIA

SIF3

SNP

TLV

Sterg

Nu sterg

Vm Record: 1/2 Exclusive Ins Num Caps

Analiza fundam Analiza tehnică Analiza v.m. V.M. Utilizatori Iesire

Selectati valoarea mobiliara pentru care se adauga date:

Descriere: SIF 3 TRANSILVANIA

SIF3

SNP

Introduceti data:

Introduceti pretul mediu:

Introduceti pretul de deschidere:

Introduceti pretul de inchidere:

Introduceti volumul tranzactiilor:

Introduceti valoarea tranzactiilor:

Introduceti numarul tranzactiilor:

Salvez

Nu salvez

Vm Record: 1/2 Exclusive Ins Num Caps

Anexa 15 -Modulele aplicației “StockAssist”

Analiza fundam Analiza tehnică Analiza v.m. V.M. Utilizatori Iesire

Selectati valoarea mobiliara pentru care se adauga date:

Descriere: **SNP PETROM**

Introduceti anul:

Introduceti profitul net: (mii lei)

Introduceti suma dividendelor (total): (mii lei)

Introduceti numarul actiunilor emise: (mii lei)

Introduceti valoarea nominala:

Introduceti cifra de afaceri: (mii lei)

Introduceti activul net: (mii lei)

Introduceti activul total: (mii lei)

Se tranzactioneaza in Bursa (T/F):

Salvez **Nu salvez**

SIF3
SNP

Vm Record: 2/2 Exclusive Ins Num

Analiza fundam Analiza tehnică Analiza v.m. V.M. Utilizatori Iesire

Ati fost introdus in lista utilizatorilor

Introduceti codul utilizatorului:

Introduceti parola:

Reconfirmati parola:

Salvez **Nu salvez**

Ins Num Caps

Anexa 15 -Modulele aplicației “StockAssist”

Analiza fundam	Analiza tehnică	Analiza v.m.	V.M.	Utilizatori	Iesire
----------------	-----------------	--------------	------	-------------	--------

Utilizatorul a fost sters din lista

Introduceti codul utilizatorului ce va fi sters din lista:

Introduceti parola:

Utilizat	Record: 1/2	Exclusive	Ins	Num	
----------	-------------	-----------	-----	-----	--

Analiza fundam	Analiza tehnică	Analiza v.m.	V.M.	Utilizatori	Iesire
----------------	-----------------	--------------	------	-------------	--------

Introduceti codul utilizatorului:

Introduceti parola:

Introduceti noua parola:

Reintroduceti noua parola:

Parola a fost modificata.

	Ins	Num	Caps
--	-----	-----	------

Anexa 16 –Codul sursă (modulul principal) al aplicației “StockAssist”

* *

* * Autor: Cristian Bologa

* *

```
public agent,var
hide window command
set talk off
set date to brit
f=capsl(.T.)
on escape close all
close all
clear
@ 10,5 say "Programul functioneaza corect doar daca
astazi este "
zi=dow(date())
do case
    case zi=1
        azi="Duminica"
    case zi=2
        azi="Luni"
    case zi=3
        azi="Marti"
    case zi=4
        azi="Miercuri"
    case zi=5
        azi="Joi"
    case zi=6
        azi="Vineri"
    case zi=7
        azi="Sambata"
endcase
@ 10,57 say azi
@ 10,67 say date()
@ 11,30 say "Este corect?"
cor="D"
@ 11,43 get cor
read
if cor<>"D"
! date
endif
clear

use utilizator

if eof()
@ 8,10 say "Programul este utilizat pentru prima oara!"
@ 10,10 say "Urmeaza sa fie introdus primul utilizator"
@ 16,28 say "Tastati-va codul: "
agent=" "
@ 16,47 get agent
read
@ 18,28 say "Parola: "
pr=" "
@ 18,40 get pr colo R/R, R/R
read
@ 20,28 say "Reconfirmati parola: "
re=" "
@ 20,50 get re colo R/R, R/R
```

```
read
if pr<>re
wait wind "Ati gresit la reconfirmare"
quit
endif
appe blank
decr=alltrim(pr)
i=1
encr=""
do while i<=len(decr)
j=asc(substr(decr,i,1))
j=j+11
j=chr(j)
encr=encr+j
i=i+1
enddo
replace cod with agent, p with encr
else
@ 11,26 to 14,50
@ 12,28 say "Tastati-va codul: "
agent=" "
@ 12,47 get agent
read
@ 13,28 say "Parola: "
pr=" "
@ 13,40 get pr colo R/R, R/R
read
use utilizator
locate for cod=agent
if .not.found()
wait wind "Cod necunoscut"
quit
endif

decr=alltrim(pr)
i=1
encr=""
do while i<=len(decr)
j=asc(substr(decr,i,1))
j=j+11
j=chr(j)
encr=encr+j
i=i+1
enddo
encr=encr+" "
encr=substr(encr,1,10)

if p<>encr
wait wind "Parola gresita"
quit
endif

endif
on key label esc do escape
close all
SET SYSMENU TO
SET SYSMENU AUTOMATIC
```

```

DEFINE PAD _rkx01ygmb OF _MSYSMENU
PROMPT "Analiza \<fundam" COLOR SCHEME 3 ;
KEY ALT+F, "ALT+F"
DEFINE PAD _rkx01ygmh OF _MSYSMENU
PROMPT "Analiza \<tehnica" COLOR SCHEME 3 ;
KEY ALT+T, "ALT+T"
DEFINE PAD _rkx01ygmo OF _MSYSMENU
PROMPT "\<Analiza v.m." COLOR SCHEME 3 ;
KEY ALT+A, "ALT+A"
DEFINE PAD _rkx01ygmV OF _MSYSMENU
PROMPT "\<V.M." COLOR SCHEME 3 ;
KEY ALT+V, "ALT+V"
DEFINE PAD _rkx01ygn0 OF _MSYSMENU
PROMPT "\<Utilizatori" COLOR SCHEME 3 ;
KEY ALT+U, "ALT+U"
DEFINE PAD _rkx01ygn6 OF _MSYSMENU
PROMPT "\<Iesire" COLOR SCHEME 3 ;
KEY ALT+I, "ALT+I"
ON PAD _rkx01ygmb OF _MSYSMENU ACTIVATE
POPUP analizafun
ON PAD _rkx01ygmh OF _MSYSMENU ACTIVATE
POPUP analizateh
ON PAD _rkx01ygmo OF _MSYSMENU ACTIVATE
POPUP analizavm
ON PAD _rkx01ygmV OF _MSYSMENU ACTIVATE
POPUP vm
ON PAD _rkx01ygn0 OF _MSYSMENU ACTIVATE
POPUP utilizator
ON SELECTION PAD _rkx01ygn6 OF _MSYSMENU
;
DO _rkx01ygnv ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")

DEFINE POPUP analizafun MARGIN RELATIVE
SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF analizafun PROMPT "Indicatori"
DEFINE BAR 2 OF analizafun PROMPT "Analiza
statistica"
DEFINE BAR 3 OF analizafun PROMPT
"Recomandari"
ON SELECTION BAR 1 OF analizafun ;
DO _rkx01ygp3 ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 2 OF analizafun ;
DO _rkx01ygpl ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 3 OF analizafun ;
DO _rkx01ygrg ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")

DEFINE POPUP analizateh MARGIN RELATIVE
SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF analizateh PROMPT "Analiza
grafica"
DEFINE BAR 2 OF analizateh PROMPT "Analiza
bazata pe EMH (random walk)"

```

```

ON SELECTION BAR 1 OF analizateh ! /n
c:\msoffice\excel\excel.exe c:\fpw26\angraf.xls
ON SELECTION BAR 2 OF analizateh ! /n
c:\msoffice\excel\excel.exe c:\fpw26\anemh.xls

DEFINE POPUP analizavm MARGIN RELATIVE
SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF analizavm PROMPT "Volatilitatea
actiunilor"
DEFINE BAR 2 OF analizavm PROMPT "Riscul
actiunilor"
DEFINE BAR 3 OF analizavm PROMPT "Valoarea
actualizata a obligatiunilor"
DEFINE BAR 4 OF analizavm PROMPT "Volatilitatea
obligatiunilor"
DEFINE BAR 5 OF analizavm PROMPT "Durata
obligatiunilor"
DEFINE BAR 6 OF analizavm PROMPT
"Recomandari"
ON SELECTION BAR 1 OF analizavm ;
DO _rkx01ygtH ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 2 OF analizavm ;
DO _rkx01ygu3 ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 3 OF analizavm ;
DO _rkx01ygtL ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 4 OF analizavm ;
DO _rkx01ygv0 ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 5 OF analizavm ;
DO _rkx01ygvG ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 6 OF analizavm ;
DO _rkx01ygt7 ;
IN LOCFILE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")

DEFINE POPUP vm MARGIN RELATIVE SHADOW
COLOR SCHEME 4
DEFINE BAR 1 OF vm PROMPT "Adaugare valori
mobiliare la lista"
DEFINE BAR 2 OF vm PROMPT "Stergere valori
mobiliare din lista"
DEFINE BAR 3 OF vm PROMPT "Introducere cursuri
valori mobiliare"
DEFINE BAR 4 OF vm PROMPT "Modificare cursuri
valori mobiliare"
DEFINE BAR 5 OF vm PROMPT "Introducere date
(anual)"
DEFINE BAR 6 OF vm PROMPT "Modificare date"
DEFINE BAR 7 OF vm PROMPT "Import date"
DEFINE BAR 8 OF vm PROMPT "Situatii"
ON SELECTION BAR 1 OF vm do adaugvm.spr

```

```

ON SELECTION BAR 2 OF vm do stergvm.spr
ON SELECTION BAR 3 OF vm do intrdate.spr
ON SELECTION BAR 4 OF vm do modifdat.spr
ON SELECTION BAR 5 OF vm do intran.spr
ON SELECTION BAR 6 OF vm do modifan.spr
ON SELECTION BAR 7 OF vm ;
    DO _rkx01ygy6 ;
    IN LOCFE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 8 OF vm ;
    DO _rkx01ygyn ;
    IN LOCFE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")

DEFINE POPUP utilizator MARGIN RELATIVE
SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF utilizator PROMPT "Adaugare
utilizatori"
DEFINE BAR 2 OF utilizator PROMPT "Stergere
utilizatori"
DEFINE BAR 3 OF utilizator PROMPT "Listare
utilizatori"
DEFINE BAR 4 OF utilizator PROMPT "Modificare
parola utilizatori"
ON SELECTION BAR 1 OF utilizator ;
    DO _rkx01ygzv ;
    IN LOCFE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 2 OF utilizator ;
    DO _rkx01yh0a ;
    IN LOCFE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 3 OF utilizator ;
    DO _rkx01yh0p ;
    IN LOCFE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")
ON SELECTION BAR 4 OF utilizator ;
    DO _rkx01yh15 ;
    IN LOCFE("PRGWIN\BURSA"
,"MPX;MPR|FXP;PRG" ,"Where is BURSA?")

clear

PROCEDURE _rkx01ygnv
clear all
close all
deactivate window all
do meniu.mpr

PROCEDURE _rkx01ygp3
close all
clear
use vm index vm
define popup subm2 prompt field simbol scroll margin
@10,1 say "Selectati valoarea mobiliara dorita: "
var=1
@4,42 get var popup subm2 function "&" size 14,16
read
fisier=simbol
Pret_curent=0

```

```

m.anul=Year(Date())-1
@19,1 say "Introduceti pretul pentru care doriti calculul
indicatorilor: "
@20,1 say "Introduceti anul pentru care doriti calculul
indicatorilor: "
@19,65 get Pret_curent picture '9999999'
read
@20,65 get m.anul picture '99999'
read
if Pret_curent=0
clear
wait wind "Indicatorii nu se pot calcula pentru un pret
curent=0"
clear
close all
return to master
endif

fisier="g"+fisier
use &fisier index &fisier
locate for anul=m.anul
if (eof()).and.(m.anul<>anul)
clear
wait wind "Inregistrarea corespunzatoare anului
introdus nu a fost gasita"
clear all
close all
return to master
endif
bvps=1000*Activ_n/Nr_act
pbv=Pret_curent/bvps
eps=1000*Profit_net/Nr_act
dvps=1000*Dividend/Nr_act
dy=100*dvps/Pret_curent
per=Pret_curent/eps
ca_act=1000*C_a/Nr_act
psr=Pret_curent/ca_act
cap_piata=Nr_act*Pret_curent/1000
roa=100*Profit_net/Activ_t
roe=100*Profit_net/Activ_n
rata_div=dvps/eps

clear
set textmerge on
defi wind w from 1,1 to 23,80
delete file rap.txt
clear
set textmerge to rap.txt noshow
\ VN -Valoare nominala:
<<Val_nom>> lei
\ BVPS -Valoare contabila:
<<bvps>> lei
\ P -Pret de piata:
<<Pret_curent>> lei
\ PN -Profit net:
<<Profit_net>> mii
lei
\ DT -Dividend total:
<<Dividend>> mii
lei

```

```

\ NA -Numar actiuni:
    <<Nr_act>> buc
\ CA -Cifra de afaceri:
    <<C_a>> mii lei
\ AN -Activ net:
    <<Activ_n>> mii lei
\ AT -Activ total:
    <<Activ_t>> mii lei
\ P/BV -Raport Pret/Valoare contabila:
    <<pbv>>
\ EPS -Profit net per actiune:
    <<eps>> lei
\ DVPS -Dividend per actiune:
    <<dvps>> lei
\ DY -Rentabilitate curenta:
    <<dy>> %
\ PER -Raport Pret/Profit net per actiune:
    <<per>>
\ CAPA -Cifra de afaceri la o actiune:
    <<ca_act>> lei
\ PSR -Raport Pret/Cifra de afaceri la o actiune:
    <<psr>>
\ CP -Capitalizare de piata:
    <<cap_piata>> mii
lei
\ ROA -Profit net/Activ total:
    <<roa>> %
\ ROE -Profit net/Capitaluri proprii:
    <<roe>> %
\ RPD -Rata de plata a dividendelor:
    <<rata_div>>

close all
modi file rap.txt noedit wind w
use
clear

PROCEDURE _rkx01ygpl
close all
clear
use vm index vm
define popup subm2 prompt field simbol scroll margin
@12,1 say "Selectati valoarea mobiliara dorita: "
var=1
@4,42 get var popup subm2 function "&" size 18,16
read
fisier=simbol
d1={ / / }
d2=date()
clear
@ 2,5 say "Perioada de calcul incepe la: "
@ 2,35 get d1
@ 4,5 say "si se termina la: "
@ 4,35 get d2
read
use &fisier index &fisier
set filter to (data>=d1).and.(data<=d2)
min=Pretmediu
max=Pretmediu
suma=0
nrart=0
disp=0
cov=0
if eof()
wait wind "Baza de date nu contine nici o inregistrare"
clear
close all
return to master
endif
do while .not.eof()
suma=suma+Pretmediu
nrart=nrart+1
if Pretmediu<min
min=Pretmediu
endif
if Pretmediu>max
max=Pretmediu
endif
skip
enddo
media=suma/nrart
go top
do while .not.eof()
disp=disp+(Pretmediu-media)*(Pretmediu-media)
skip
enddo
amp=sqrt(disp)
if media<>0
coefpears=amp/media
endif
set textmerge on
defi wind w from 1,1 to 23,80
delete file rap.txt
clear
set textmerge to rap.txt noshow

\
\
\ indicatorilor statistici
\ pentru perioada cuprinsa intre
<<d1>> si <<d2>>
\
\ Valoarea minima:
    <<min>>
\ Valoarea maxima:
    <<max>>
\ Valoarea medie:
    <<media>>
\ Dispersia:
    <<disp>>
\ Abaterea medie patratica:
    <<amp>>
\ Coeficientul de variatie a lui Pearson:
    <<coefpears>>

close all
modi file rap.txt noedit wind w
use

```

```

clear

PROCEDURE _rnx01ygrg
clear
close all
dimension c[8]
store 0 to c
defi wind w from 1,1 to 23,80
activate window w
? 'In analiza fundamentala se folosesc mai multi
indicatori. Fiecare poate'
? 'sa influenteze decizia de vanzare sau de cumparare.
Unii indicatori'
? 'prezinta o importanta mai mare. De aceea e nevoie de
o ierarhizare a'
? 'lor. Fiecaruia din indicatorii de mai jos i se va asocia
un coeficient'
? 'de importanta (0-100):'
? '1. PBV=Raportul Valoare de piata / Valoare
contabila'
? '2. EPS=Profit net per actiune'
? '3. DY=Rentabilitate curenta'
? '4. PER=Raportul Pret/ Castig'
? '5. PSR=Raportul Pret/ Cifra de afaceri la o actiune'
? '6. RPD=Rata de plata a dividendelor'
? '7. ROA=Raportul Profit net / Activ total'
? '8. ROE=Raportul Profit net / Active nete'
@6,55 get c[1] picture '999'
@7,55 get c[2] picture '999'
@8,55 get c[3] picture '999'
@9,55 get c[4] picture '999'
@10,55 get c[5] picture '999'
@11,55 get c[6] picture '999'
@12,55 get c[7] picture '999'
@13,55 get c[8] picture '999'
read
clear
m.anul=Year(Date())-1
nrani=0
@ 2,1 say "Introduceti numarul de ani luat in calcul: "
@ 2,65 get nrani picture '99'
@ 3,1 say "Introduceti anul cu care se incheie perioada: "
@ 3,65 get m.anul picture '99999'
read
deactivate window w

use vm index vm
count to nrelem
go top
dimension
descr[nrelem],pbv[nrelem,nrani],eps[nrelem,nrani],dy[nrelem,nrani]
dimension
per[nrelem,nrani],psr[nrelem,nrani],rpd[nrelem,nrani]
dimension
roa[nrelem,nrani],roe[nrelem,nrani],rep[nrelem]
store 0 to rep
contor=0

```

```

do while .not.eof()
contor=contor+1
descr[contor]=descriere
fisier=simbol
gfisier="g"+simbol
select 2
for i=1 to nrani
use &fisier index &fisier
do while (.not.eof()).and.(year(Data)<>(m.anul-i+1))
skip
enddo
if .not.eof()
Pret_curent=Pretmediu
else
Pret_curent=0
endif
use &gfisier index &gfisier
locate for anul=m.anul-i+1
if found()
bvps=1000*Activ_n/Nr_act
pbv[contor,i]=Pret_curent/bvps
eps[contor,i]=1000*Profit_net/Nr_act
dvps=1000*Dividend/Nr_act
if Pret_curent<>0
dy[contor,i]=100*dvps/Pret_curent
else
dy[contor,i]=0
endif
per[contor,i]=Pret_curent/eps[contor,i]
ca_act=1000*C_a/Nr_act
psr[contor,i]=Pret_curent/ca_act
roa[contor,i]=100*Profit_net/Activ_t
roe[contor,i]=100*Profit_net/Activ_n
rpd[contor,i]=dvps/eps[contor,i]
else
pbv[contor,i]=0
eps[contor,i]=0
dy[contor,i]=0
per[contor,i]=0
psr[contor,i]=0
roa[contor,i]=0
roe[contor,i]=0
rpd[contor,i]=0
endif
endfor
select 1
skip
enddo

*calcul ponderi
suma=0
for i=1 to 8
suma=suma+c[i]
endfor
for i=1 to 8
c[i]=100*c[i]/suma
endfor

close all
clear

```

```

set textmerge on
defi wind w from 1,1 to 33,80
delete file rap.txt
clear
set textmerge to rap.txt noshow

```

```

* analiza pbv
\ Evolutia PBV (Raport valoare de piata/ Valoare
contabila):
minpbv=pbv[1,1]
ipbv=1
for contor=1 to nrelem
if pbv[contor,1]<minpbv
ipbv=contor
minpbv=pbv[contor,1]
endif
endfor
rep[ipbv]=rep[ipbv]+c[1]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if pbv[contor,i]<>0
\ PBV in anul <<(m.anul-i+1)>>: <<pbv[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[ipbv]>>
\

```

```

* analiza eps
\ Evolutia EPS (Profit net per actiune):
maxeps=eps[1,1]
ieps=1
for contor=1 to nrelem
if eps[contor,1]>maxeps
ieps=contor
maxeps=eps[contor,1]
endif
endfor
rep[ieps]=rep[ieps]+c[2]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if eps[contor,i]<>0
\ EPS in anul <<(m.anul-i+1)>>: <<eps[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[ieps]>>
\

```

```

* analiza dy

```

```

\ Evolutia DY (Rentabilitate curenta):
maxdy=dy[1,1]
idy=1
for contor=1 to nrelem
if dy[contor,1]>maxdy
idy=contor
maxdy=dy[contor,1]
endif
endfor
rep[idy]=rep[idy]+c[3]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if dy[contor,i]<>0
\ DY in anul <<(m.anul-i+1)>>: <<dy[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[idy]>>
\

```

```

* analiza per
\ Evolutia PER (Raportul Pret/Profit net):
minper=per[1,1]
iper=1
for contor=1 to nrelem
if per[contor,1]<minper
iper=contor
minper=per[contor,1]
endif
endfor
rep[iper]=rep[iper]+c[4]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if per[contor,i]<>0
\ PER in anul <<(m.anul-i+1)>>: <<per[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[iper]>>
\

```

```

* analiza psr
\ Evolutia PSR (Raportul Pret/Cifra de afaceri la o
actiune):
minpsr=psr[1,1]
ipsr=1
for contor=1 to nrelem
if psr[contor,1]<minpsr
ipsr=contor
minpsr=psr[contor,1]

```

```

endif
endfor
rep[ipsr]=rep[ipsr]+c[5]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if psr[contor,i]<>0
\ PSR in anul <<(m.anul-i+1)>>: <<psr[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[ipsr]>>
\

* analiza rpd
\ Evolutia RPD (Rata de plata a dividendelor):
maxrpd=rp[1,1]
irpd=1
for contor=1 to nrelem
if rpd[contor,1]>maxrpd
irpd=contor
maxrpd=rp[contor,1]
endif
endfor
rep[irpd]=rep[irpd]+c[6]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if rpd[contor,i]<>0
\ RPD in anul <<(m.anul-i+1)>>: <<rpd[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[irpd]>>
\

* analiza roa
\ Evolutia ROA (Raportul Profit net/Activ total):
maxroa=roa[1,1]
iroa=1
for contor=1 to nrelem
if roa[contor,1]>maxroa
iroa=contor
maxroa=roa[contor,1]
endif
endfor
rep[iroa]=rep[iroa]+c[7]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if roa[contor,i]<>0

```

```

\ ROA in anul <<(m.anul-i+1)>>: <<roa[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[iroa]>>
\

* analiza roe
\ Evolutia ROE (Raportul Profit net/Active nete):
maxroe=roe[1,1]
iroe=1
for contor=1 to nrelem
if roe[contor,1]>maxroe
iroe=contor
maxroe=roe[contor,1]
endif
endfor
rep[iroe]=rep[iroe]+c[8]
for contor=1 to nrelem
\
\ Societatea <<descr[contor]>>
for i=1 to nrani
if roe[contor,i]<>0
\ ROE in anul <<(m.anul-i+1)>>: <<roe[contor,i]>>
endif
endfor
endfor
\
\ Din punct de vedere al acestui indicator va
recomandam investitia in
\ actiuni <<descr[iroe]>>
\
-----
-
\ Conform criteriilor formulate de dvs, portofoliul optim
consta in:
\
for contor=1 to nrelem
\ <<rep[contor]>>% actiuni <<descr[contor]>>
endfor
\
-----
-
close all
modi file rap.txt noedit wind w
use
clear

PROCEDURE _rkx01ygh
close all
clear
m.anul=Year(Date())-1
nrani=0
@ 2,1 say "Introduceti numarul de ani luat in calcul: "
@ 2,65 get nrani picture '99'
@ 3,1 say "Introduceti anul cu care se incheie perioada: "
@ 3,65 get m.anul picture '99999'

```

```

read

use vm index vm
define popup subm2 prompt field simbol scroll margin
@10,1 say "Selectati valoarea mobiliara dorita: "
var=1
@5,42 get var popup subm2 function "&" size 14,16
read
alegere=simbol
den=descriere

* calculeaza rentabilitatea de piata
dimension sumarentab[nrani]
dimension nrrentab[nrani]
dimension rentab[nrani]
dimension srindiv[nrani]
dimension nrrindiv[nrani]
dimension rindiv[nrani]
store 0 to sumarentab
store 0 to nrrentab
store 0 to rentab
store 0 to srindiv
store 0 to nrrindiv
store 0 to rindiv

for i=1 to nrani
use vm index vm
do while .not.eof()
fisier=simbol
gfisier="g"+simbol
select 2
use &fisier index &fisier
do while (.not.eof()).and.(year(Data)<>(m.anul-i+1))
skip
enddo
if .not.eof()
Pretcurent=Pretmediu
else
Pretcurent=0
endif
use &gfisier index &gfisier
locate for anul=m.anul-i+1
if found()
dvps=1000*Dividend/Nr_act
else
dvps=0
endif
if (Pretcurent<>0).and.(dvps<>0)
sumarentab[i]=sumarentab[i]+100*dvps/Pretcurent
nrrentab[i]=nrrentab[i]+1
endif
if (alegere=fisier).and.(Pretcurent<>0).and.(dvps<>0)
srindiv[i]=srindiv[i]+100*dvps/Pretcurent
nrrindiv[i]=nrrindiv[i]+1
endif
sele 1
skip
enddo
endfor

for i=1 to nrani
if nrrentab[i]<>0
rentab[i]=sumarentab[i]/nrrentab[i]
endif
if nrrindiv[i]<>0
rindiv[i]=srindiv[i]/nrrindiv[i]
endif
endfor

* calculeaza alfa si beta
sri=0
srp=0
srpp=0
srirp=0
n=0
for i=1 to nrani
if (rentab[i]<>0).and.(rindiv[i]<>0)
sri=sri+rindiv[i]
srp=srp+rentab[i]
srpp=srpp+rentab[i]*rentab[i]
srirp=srirp+rindiv[i]*rentab[i]
n=n+1
endif
endfor
alfa=(sri*srpp-srp*srirp)/(n*srpp-srp*srp)
beta=(n*srirp-sri*srp)/(n*srpp-srp*srp)

close all
clear
set textmerge on
defi wind w from 1,1 to 26,80
delete file rap.txt
clear
set textmerge to rap.txt noshow
\ Societate: <<den>>
for i=1 to nrani
if rindiv[i]<>0
\ Rentab. individuala in anul <<(m.anul-i+1)>>:
<<rindiv[i]>>
\ Rentab. pietei in anul <<(m.anul-i+1)>>:
<<rentab[i]>>
endif
endfor
\
\ Conform metodei celor mai mici patrate, alfa si beta au
urmatoarele valori:
\ Alfa=<<alfa>>
\ Beta=<<beta>>
\ Ecuatia dreptei de regresie este:
\ ri=<<alfa>>+<<beta>>rp
\ unde ri este rentabilitatea individuala a actiunii alese
\ iar rp este rentabilitatea generala a pietei.
\
\ Alfa semnifica intersectia dreptei de regresie cu axa
ordonatelor si reprezinta rentabilitatea actiunii
respective atunci cand rentabilitatea pietei este nula.
\ Beta este coeficientul de regresie sau volatilitatea
(numit si coeficient beta).
close all
modi file rap.txt noedit wind w

```

```

use
clear

PROCEDURE _rkx01ygu3
close all
clear
m.anul=Year(Date())-1
nrani=0
@ 2,1 say "Introduceti numarul de ani luat in calcul: "
@ 2,65 get nrani picture '99'
@ 3,1 say "Introduceti anul cu care se incheie perioada: "
"

@ 3,65 get m.anul picture '99999'
read

use vm index vm
define popup subm2 prompt field simbol scroll margin
@10,1 say "Selectati valoarea mobiliara dorita: "
var=1
@5,42 get var popup subm2 function "&" size 14,16
read
alegere=simbol
den=descriere

* calculeaza rentabilitatea de piata
dimension sumarentab[nrani]
dimension nrrentab[nrani]
dimension rentab[nrani]
dimension srindiv[nrani]
dimension nrrindiv[nrani]
dimension rindiv[nrani]
store 0 to sumarentab
store 0 to nrrentab
store 0 to rentab
store 0 to srindiv
store 0 to nrrindiv
store 0 to rindiv

for i=1 to nrani
use vm index vm
do while .not.eof()
fisier=simbol
gfisier="g"+simbol
select 2
use &fisier index &fisier
do while (.not.eof()).and.(year(Data)<>(m.anul-i+1))
skip
enddo
if .not.eof()
Pretcurent=Pretmediu
else
Pretcurent=0
endif
use &gfisier index &gfisier
locate for anul=m.anul-i+1
if found()
dvps=1000*Dividend/Nr_act
else
dvps=0
endif

```

```

if (Pretcurent<>0).and.(dvps<>0)
sumarentab[i]=sumarentab[i]+100*dvps/Pretcurent
nrrentab[i]=nrrentab[i]+1
endif
if (alegere=fisier).and.(Pretcurent<>0).and.(dvps<>0)
srindiv[i]=srindiv[i]+100*dvps/Pretcurent
nrrindiv[i]=nrrindiv[i]+1
endif
sele 1
skip
enddo
endfor

for i=1 to nrani
if nrrentab[i]<>0
rentab[i]=sumarentab[i]/nrrentab[i]
endif
if nrrindiv[i]<>0
rindiv[i]=srindiv[i]/nrrindiv[i]
endif
endfor

* calculeaza elementele matricii de variatie si covariatie
sri=0
srp=0
srip=0
srpp=0
srirp=0
n=0
for i=1 to nrani
if (rentab[i]<>0).and.(rindiv[i]<>0)
sri=sri+rindiv[i]
srp=srp+rentab[i]
srip=srip+rindiv[i]*rindiv[i]
srpp=srpp+rentab[i]*rentab[i]
srirp=srirp+rindiv[i]*rentab[i]
n=n+1
endif
endfor
m00=(n*srip-sri*sri)/n*n
m01=(n*srirp-sri*srp)/n*n
m11=(n*srpp-srp*srp)/n*n
c=m01/sqrt(m00*m11)
close all
clear
set textmerge on
defi wind w from 1,1 to 23,80
delete file rap.txt
clear
set textmerge to rap.txt noshow
\Societate: <<den>>
for i=1 to nrani
if rindiv[i]<>0
\S Rentab. individuala in anul <<(m.anul-i+1)>>:
<<rindiv[i]>>
\S Rentab. pietei in anul <<(m.anul-i+1)>>:
<<rentab[i]>>
endif
endfor
\

```

```

\ Coeficientul de corelatie este: <<c>>
\
\ Coeficientul de corelatie masoara gradul de dependenta
al actiunii
\selectate fata de piata. Pentru valori pozitive avem o
dependenta direct
\proportionala intre actiunea selectata si piata bursiera
iar pentru valori
\negative relatia de dependenta este invers
proportionala. Pentru C=0 nu
\exista nici o legatura intre rentabilitatea individuala si
cea de piata.
\ Pentru C=1, variatia rentabilitatii actiunii este integral
si exclusiv
\datorata variatiei rentabilitatii generale a pietei bursiere
(cea mai mare
\volatilitate si risc specific 0).
close all
modi file rap.txt noedit wind w
use
clear

PROCEDURE _rkx01ygv0
clear
close all
defi wind w from 1,1 to 23,80
activate window w
input ' Introduceti numarul perioadelor de calcul: ' to nrc
rez=0
putere=1
for i=1 to nrc
?
? " Perioada ",i
input ' Introduceti valoarea: ' to valoare
input ' Introduceti factorul de actualizare: ' to fact
putere=putere*(1+fact)
rez=rez+valoare/putere
endfor
?
? " Valoarea prezenta a acestor fluxuri de numerar este:
",rez picture '99999999.9'
wait ""
deactivate window w
clear

PROCEDURE _rkx01ygv0
clear
close all
defi wind w from 1,1 to 23,80
activate window w
input ' Introduceti valoarea nominala a obligatiunii: ' to
valnom
vp=0
putere=1
for t=1 to nrc
?
? " Perioada ",t
input ' Introduceti rata scadenta din capital: ' to cap
input ' Introduceti dobanda incasata: ' to dob

```

```

input ' Introduceti factorul de actualizare: ' to fact
putere=putere*(1+fact)
vp=vp+(cap+dob)/putere
endfor
vol=100*abs(vp-valnom)/valnom
?
? " Volatilitatea acestei obligatiuni este: ",vol picture
'99.99','%'
wait ""
deactivate window w
clear

PROCEDURE _rkx01ygv7
clear
close all
defi wind w from 1,1 to 23,80
activate window w
input ' Introduceti valoarea nominala a obligatiunii: ' to
vn
input ' Introduceti numarul perioadelor de calcul: ' to nrc
dur=0
pv=0
putere=1
for t=1 to nrc
?
? " Perioada ",t
input ' Introduceti rata scadenta din capital: ' to cap
input ' Introduceti dobanda incasata: ' to dob
input ' Introduceti factorul de actualizare: ' to fact
putere=putere*(1+fact)
pv=pv+(cap+dob)/putere
dur=dur+(cap+dob)*t/putere
endfor
dur=dur/pv
?
? " Durata acestei obligatiuni (Duration) este: ",dur
picture '9999.99'
wait ""
deactivate window w
clear

PROCEDURE _rkx01ygv7
clear
close all
defi wind w from 1,1 to 23,80
activate window w
? ' In functie de coeficientul beta, actiunile se
clasifica in:
? '1)actiuni foarte volatile. Acestea au beta>1; o crestere
(scadere) mai
? 'mare de 1% a indicelui general al pietei bursiere
provoaca o crestere
? '(scadere) mai mare de 1% a pretului respectivelor
actiuni.'
? '2)actiuni putin volatile. Acestea au beta<1; o crestere
(scadere) mai
? 'mare de 1% a indicelui general al pietei bursiere
provoaca o crestere
? '(scadere) mai mica de 1% a pretului respectivelor
actiuni.'

```

```

? '3)actiuni cu beta=1. Evolueaza in acelasi ritm cu
piata.'
? '      Daca doriti o politica agresiva constand in
maximizarea profitului'
? 'va trebui sa va orientati spre actiuni din prima grupa;
daca urmariti o'
? 'politica defensiva, de conservare a capitalului si
reducere a riscului,'
? 'se recomanda investitia in actiuni din a treia grupa.'
?
? ' In portofoliul dvs ce pondere(0-100) doriti sa
acordati:'
input 'Categoriei 1): ' to opt1p1
input 'Categoriei 2): ' to opt1p2
input 'Categoriei 3): ' to opt1p3
clear
? '      Variabilitatea totala a rentabilitatii unui titlu
are 2 componente:'
? '1)risc sistematic. Este determinat de influenta pietei
bursiere. El se'
? 'refera la partea din variabilitatea totala cauzata de
factori ce'
? 'afecteaza pretul tuturor titlurilor.'
? '2)risc specific. E determinat de influenta
caracteristicilor specifice'
? 'ale fiecarui titlu in parte. Riscul specific e partea din
riscul total'
? 'ce e unic pentru o firma sau o industrie.'
? '      Riscul se masoara prin coeficientul de corelatie
C care masoara'
? 'gradul de dependenta al actiunii fata de piata. O
investitie este cu'
? 'atat mai riscanta cu cat are un risc specific mai
ridicat.'
? ' In portofoliul dvs ce pondere(0-100) doriti sa
acordati:'
input 'Categoriei 1): ' to opt2p1
input 'Categoriei 2): ' to opt2p2
clear
deactivate window w

m.anul=Year(Date())-1
nrani=0
@ 2,1 say "Introduceti numarul de ani luat in calcul: "
@ 2,65 get nrani picture '99'
@ 3,1 say "Introduceti anul cu care se incheie perioada:
"
@ 3,65 get m.anul picture '99999'
read
use vm index vm
count to nrelem
store " to descr
store 0 to rt
contor=0

* calculeaza rentabilitatea fiecarei valori mobiliare
dimension descr[nrelem]
dimension rt[nrelem,nrani]
dimension rentabpiata[nrani]
dimension alfa[nrelem]

```

```

dimension beta[nrelem]
dimension C[nrelem]
go top
do while .not.eof()
contor=contor+1
descr[contor]=descriere
fisier=simbol
gfsier="g"+simbol
select 2

for i=1 to nrani
store 0 to srindiv
store 0 to nrrindiv
use &fisier index &fisier
do while (.not.eof()).and.(year(Data)<>(m.anul-i+1))
skip
enddo
if .not.eof()
Pretcurent=Pretmediu
else
Pretcurent=0
endif
use &gfsier index &gfsier
locate for anul=m.anul-i+1
if found()
dvps=1000*Dividend/Nr_act
else
dvps=0
endif
if (Pretcurent<>0).and.(dvps<>0)
srindiv=srindiv+100*dvps/Pretcurent
nrrindiv=nrrindiv+1
endif
if nrrindiv<>0
rt[contor,i]=srindiv/nrrindiv
else
rt[contor,i]=0
endif
endfor
sele 1
skip
enddo

store 0 to rentabpiata
for i=1 to nrani
srentabpiata=0
nrrentabpiata=0
for contor=1 to nrelem
if rt[contor,i]<>0
srentabpiata=srentabpiata+rt[contor,i]
nrrentabpiata=nrrentabpiata+1
endif
endfor
rentabpiata[i]=srentabpiata/nrrentabpiata
endfor

*calculeaza alfa, beta si coeficientul de corelatie
for contor=1 to nrelem
sri=0
srp=0

```

```

srip=0
srpp=0
srirp=0
n=0
for i=1 to nrani
if (rt[contor,i]<>0).and.(rentabpiata[i]<>0)
sri=sri+rt[contor,i]
srp=srp+rentabpiata[i]
srip=srip+rt[contor,i]*rt[contor,i]
srpp=srpp+rentabpiata[i]*rentabpiata[i]
srirp=srirp+rt[contor,i]*rentabpiata[i]
n=n+1
endif
endfor
alfa[contor]=(sri*srpp-srp*srirp)/(n*srpp-srp*srp)
beta[contor]=(n*srirp-sri*srp)/(n*srpp-srp*srp)
m00=(n*srip-sri*n)/n
m01=(n*srirp-sri*srp)/n
m11=(n*srpp-srp*srp)/n
C[contor]=m01/sqrt(m00*m11)
endfor *contor

```

```

o1p1=100*(opt1p1+opt1p3/2)/(opt1p1+opt1p2+opt1p3)
o1p2=100*(opt1p2+opt1p3/2)/(opt1p1+opt1p2+opt1p3)
o2p1=100*opt2p1/(opt2p1+opt2p2)
o2p2=100*opt2p2/(opt2p1+opt2p2)

```

```

minbeta=beta[1]
iminbeta=1
maxbeta=beta[1]
imaxbeta=1
mincoefc=C[1]
imincoefc=1
maxcoefc=C[1]
imaxcoefc=1
for contor=2 to nrelem
if beta[contor]<minbeta
minbeta=beta[contor]
iminbeta=contor
endif
if C[contor]<mincoefc
mincoefc=C[contor]
imincoefc=contor
endif
if beta[contor]>maxbeta
maxbeta=beta[contor]
imaxbeta=contor
endif
if C[contor]>maxcoefc
maxcoefc=C[contor]
imaxcoefc=contor
endif
endfor

```

```

close all
clear
set textmerge on
defi wind w from 1,0 to 33,79
delete file rap.txt
clear

```

```

set textmerge to rap.txt noshow
\ Rentabilitatea de piata:
for i=1 to nrani
if rentabpiata[i]<>0
\ In anul <<(m.anul-i+1)>>: <<rentabpiata[i]>>
endif
endfor
for contor=1 to nrelem
\
\ Societate: <<descr[contor]>>
for i=1 to nrani
if rt[contor,i]<>0
\ Rentab. individuala in anul <<(m.anul-i+1)>>:
<<rt[contor,i]>>
endif
endfor
\ Alfa=<<alfa[contor]>>
Beta=<<beta[contor]>>
\ Coeficientul de corelatie C=<<c[contor]>>
endfor
\
\
\ Din punctul de vedere al primului criteriu va
recomandam o investitie
\ constand din:
\ <<o1p1>>% de investit in actiuni
<<Descr[imaxbeta]>>
\ <<o1p2>>% de investit in actiuni
<<Descr[iminbeta]>>
\
\ Din punctul de vedere al celui de al doilea criteriu va
recomandam o
\ investitie constand in:
\ <<o2p1>>% de investit in actiuni
<<Descr[imaxcoefc]>>
\ <<o2p2>>% de investit in actiuni
<<Descr[imincoefc]>>

close all
modi file rap.txt noedit wind w
use
clear

PROCEDURE _rkx01ygy6
clear
close all
use vm index vm
define popup subm prompt field simbol scroll margin
@ 12,1 say "Selectati simbolul dorit"
var=1
@ 4,32 get var popup subm function "&" size 18,16
read
s1=simbol
s="a:\"+simbol+".dbf"
@ 22,10 say "Introduceti discheta"
! a:
use &s1 index &s1
on error @ 23,20 say "Fisierul nu a fost gasit"
appe from &s
close all

```



```

clear read

else
wait wind "Nu am salvat"
clear read
close all
clear
endif

return

PROCEDURE _rkx01yh0a
close all
clear
@ 10,5 say "Introduceti codul utilizatorului ce va fi sters din lista: "
c=" "
@ 10,70 get c valid c<>" " error "Reintroduceti"
read
@ 12,5 say "Introduceti parola:"
p1=" "
@ 12,30 get p1 colo R/R, R/R
read

decr=alltrim(p1)
i=1
encr=""
do while i<=len(decr)
j=asc(substr(decr,i,1))
j=j+11
j=chr(j)
encr=encr+j
i=i+1
enddo
encr=encr+" "
encr=substr(encr,1,10)

use utilizator
locate for cod=c

if .not. found()
wait wind "Utilizator necunoscut"
do escape
return to master
endif

if p<>encr
wait wind "Parola gresita"
do escape
return to master
endif

pu=1
@ 18,50 get pu func '*NH \!Salvez;\!Nu salvez' size 1,8,3 valid salv() defa 1
read
return

procedure salv
if pu<>1

```

```

wait wind "Nu am salvat"
clear read
else
dele
pack
wait wind "Utilizatorul a fost sters din lista"
clear read
endif
clear
close all
return

PROCEDURE _rkx01yh0p
close all
clear
use utilizator

@ 4,1 say " Lista utilizatorilor: "

define popup subm1 prompt field cod scroll margin
var=1
@4,32 get var popup subm1 function "&" size 18,16
read
use
clear

PROCEDURE _rkx01yh15
close all
clear
use utilizator
@ 10,5 say "Introduceti codul utilizatorului: "
c=" "
@ 10,40 get c valid c<>" "
read
@ 12,5 say "Introduceti parola:"
p1=" "
@ 12,30 get p1 colo R/R, R/R
read
locate for cod=c
if .not.found()
@ 20,10 say "Acest nume de utilizator nu exista in lista de utilizatori"
wait ""
use
clear
return
endif

decr=alltrim(p1)
i=1
encr=""
do while i<=len(decr)
j=asc(substr(decr,i,1))
j=j+11
j=chr(j)
encr=encr+j
i=i+1
enddo
encr=encr+" "
encr=substr(encr,1,10)

```

```

if p<>encr
@ 20,10 say "Ati gresit parola."
wait ""
use
clear
return
endif

@ 16,5 say "Introduceti noua parola:"
p1=""
@ 16,30 get p1 colo R/R, R/R
read
@ 17,5 say "Reintroduceti noua parola:"
p2=""
@ 17,30 get p2 colo R/R, R/R
read
if p1<>p2
@ 20,5 say "Ati gresit la reconfirmarea parolei. Parola
ramane nemodificata."
wait ""
use
clear
return
endif

decr=alltrim(p1)
i=1
encr=""
do while i<=len(decr)
j=asc(substr(decr,i,1))
j=j+11
j=chr(j)
encr=encr+j
i=i+1
enddo

replace p with encr
use
@ 20,10 say "Parola a fost modificata."
wait ""
clear
return

```

Anexa 17 – Codul sursă al aplicației “Expert Generator”

```

predicates
    motor1
    incarca
    descarca
    identifica
    identifica_entitate(integer,integer,integer)/*Cod_entitate*/
    identifica_caracterist(integer,integer,symbol)
    este(integer,integer,symbol)
    intreaba(symbol)
    intreaba2(symbol)
    tine_minte(integer,integer,symbol)
    /*****/
    princ
    meniu(char)
    m
    n
    numara(integer)
    men(char)
    sterge(string)
    st(char,string)
    citescentitate(integer)
    citesccaracteriste(integer)
    citescvalori(integer,integer)
    citescexclusiv(symbol,integer)
    citesccunost(integer,integer)
    citesccunost2(integer,integer)
    citesccunost3(integer,integer,integer)
    afiseazacaracterist(integer)
    afiseazavalori(integer,integer)
    adaugcunost2(integer,integer)
database
    /*entitate <nume_entitate> <cod_entitate> <nr_de_legatura>*/
    entitate(symbol,integer,integer)

    /*caracterist <nume_caracterist> <cod_caracterist> <mesaj_inainte>
<mesaj_dupa>*/
    caracterist(symbol,integer,symbol,symbol)

    /*EXCLUSIV <cod_caracterist>*/
    exclusiv(integer)

    /*VALOARE <cod_caracterist> <nume_valoare> <cod_valoare>*/
    valoare(integer,symbol,integer)

    /*legatura <cod_entitate> <cod_caracterist> <cod_valoare> codlegatura*/
    legatura(integer,integer,integer,integer)

    /*pentru precizarea tipului entitateului de identificat*/
    comentariu_entitate(symbol)

    /*BAZA TEMPORARA DIN MEMORIE*/

    /* STIE <cod_caracterist> <cod_valoare> <da/nu>*/
    stie(integer,integer,symbol)

goal
    princ.

```

Anexa 17 – Codul sursă al aplicației “Expert Generator”

clauses

[illegible]

Anexa 17 – Codul sursă al aplicației “Expert Generator”

[illegible]

Anexa 17 – Codul sursă al aplicației “Expert Generator”

[illegible]

Anexa 17 – Codul sursă al aplicației “Expert Generator”

```

    assertz(entitate(Numentitate,N,Nrlegatura)),
    N1=N-1,
    citescentitate(N1).

citesccaracteriste(0):-!.

citesccaracteriste(M):-nl,write("Introduceti numele caracteristicii:
"),
    readln(Numecaracterist),
    write("Introduceti mesaj initial: "),
    readln(Mesajinit),
    write("Introduceti mesaj final: "),
    readln(Mesajfinal),
    write("Aceasta caracteristica este exclusiva? (da/nu): "),
    readln(Excl),
    citescexclusiv(Excl,M),
    assertz(caracterist(Numecaracterist,M,Mesajinit,Mesajfinal)),
    write("Cate valori avem pentru aceasta caracteristica? "),
    readint(Nv),
    citescvalori(M,Nv),
    M1=M-1,
    citesccaracteriste(M1).

citescexclusiv(da,M):-assertz(exclusiv(M)).

citescexclusiv(nu,_).

citescvalori(_,0):-!.

citescvalori(M,Nv):-write("Introduceti numele valorii: "),
    readln(Numeval),
    assertz(valoare(M,Numeval,Nv)),
    Nv1=Nv-1,
    citescvalori(M,Nv1).

citesccunost(I,J):-
    J=1,entitate(Nume,I,Nc),
    nl,write("Dati cunostintele despre entitatea "),
    write(Nume),nl,
    citesccunost2(I,J);
    J>1,entitate(_,I,_),citesccunost2(I,J);
    not(entitate(_,I,_)).

citesccunost2(I,J):-
    write("*****"),nl,
    write("Cod caracterist"),nl,
    write("*****"),nl,
    afiseazacaracterist(1),
    write("*****"),nl,
    write("Introduceti codul caracteristicii: "),
    readint(Codat),
    nl,write("*****"),nl,
    write("Cod Valoare"),nl,
    write("*****"),nl,
    afiseazavalori(1,Codat),
    write("*****"),nl,
    write("Introduceti codul valorii: "),
    readint(Codval),
    assertz(legatura(I,Codat,Codval,J)),

```

Anexa 17 – Codul sursă al aplicației “Expert Generator”

```

entitate(_,I,Nc),
J1=J+1,
citesccunost3(I,J1,Nc).

citesccunost3(I,J1,Nc):-J1<=Nc,!,citesccunost(I,J1).

citesccunost3(I,J1,Nc):-J1>Nc,I1=I+1,citesccunost(I1,1).

afiseazacaracterist(I):-not(caracterist(_,I,_,_));
    caracterist(Numeat,I,_,_),
    write(I,"    ",Numeat),nl,
    I1=I+1,
    afiseazacaracterist(I1).

afiseazavalori(J,Codat):-not(valoare(Codat,_,J));
    valoare(Codat,Numev,J),
    write(J,"    ",Numev),nl,
    J1=J+1,
    afiseazavalori(J1,Codat).

motor1:-
    incarca,
    identifica,
    descarca.

incarca:-
    dir(".", "*.cun",Nume_cun),nl,nl,
    consult(Nume_cun),
    write("Am incarcat baza de cunostinte: ",Nume_cun,"."),nl,!;
    write("Nu am reusit sa incarc baza de cunostinte."),nl.

descarca:-retractall(_).

identifica:-
    entitate(Nume_entitate,Cod_entitate,Nr_de_legatura),nl,
    identifica_entitate(Cod_entitate,Nr_de_legatura,1),
    comentariu_entitate(Cob),
    write("Este ",Cob," ",Nume_entitate,"!"),nl.

identifica:-write("Nu stiu ce e."),nl.

identifica_entitate(Cod_entitate,Nr_de_legatura,N):-
    N=Nr_de_legatura+1;
    legatura(Cod_entitate,Cod_caracterist,Cod_valoare,N),
    identifica_caracterist(Cod_caracterist,Cod_valoare,DN),
    DN="da",Np1=N+1,
    identifica_entitate(Cod_entitate,Nr_de_legatura,Np1).

identifica_caracterist(Cod_caracterist,Cod_valoare,DN):-
    stie(Cod_caracterist,Cod_valoare,"da"),DN="da",!.

identifica_caracterist(Cod_caracterist,Cod_valoare,DN):-
    stie(Cod_caracterist,Cod_valoare,"nu"),DN="nu",!.

identifica_caracterist(Cod_caracterist,_,DN):-
    exclusiv(Cod_caracterist),stie(Cod_caracterist,_, "da"),DN="nu",!.

identifica_caracterist(Cod_caracterist,Cod_valoare,DN):-
    caracterist(_,Cod_caracterist,Mesaj_inainte,Mesaj_dupa),

```

Anexa 17 – Codul sursă al aplicației “Expert Generator”

```
valoare(Cod_caracterist,Nume_valoare,Cod_valoare),
write(Mesaj_inainte," ",Nume_valoare,Mesaj_dupa),nl,
este(Cod_caracterist,Cod_valoare,DN),
tine_minte(Cod_caracterist,Cod_valoare,DN).

este(Cod_caracterist,Cod_valoare,DN):-
    stie(Cod_caracterist,Cod_valoare,DN);
    intreaba(DN).

intreaba(DN):-write("da/nu? "),readln(DN),intreaba2(DN).

intreaba2(DN):-DN="da",nl,nl.

tine_minte(Cod_caracterist,Cod_valoare,DN):-
    assert(stie(Cod_caracterist,Cod_valoare,DN)).
```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

[illegible][illegible]

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|      +-----Meniu baza de cunostinte-----+
|      |      +-----+
|      |      |      CREARE BAZA DE CUNOSTINTE      |
|      |      +-----+
|      |      Numele bazei de cunostinte(Max 8 caractere+ .CUN): prodvinc.cun
|      |
|      |      Introduceti tipul entitatii de identificat: produsul financiar
|      |
|      |      Introduceti numarul de entitati: 8
|      |
|      |      Introduceti numele obiectului: actiune preferentiala
|      |      Introduceti nr de cunostinte pentru acest obiect: 6
|      |
|      |      Introduceti numele obiectului: actiune obisnuita
|      |      Introduceti nr de cunostinte pentru acest obiect: 7
|      |
|      |      Introduceti numele obiectului: actiune de aport
|      |
|      +-----+
|
|      ||
|      ||
|      +-----+
|
++-----+
F2-Save  F3-Load  F6-Switch  F9-Compile
Alt-X-Exit

```

```

Files          Edit          Run          Compile          Options          Setup
++-----Meniu principal-----++
|
|  +-----Meniu baza de cunostinte-----+
|  |
|  | Introduceti nr de cunostinte pentru acest obiect: 7
|  |
|  | Introduceti numele obiectului: actiune cu dividend prioritar
|  | Introduceti nr de cunostinte pentru acest obiect: 7
|  |
|  | Introduceti numele obiectului: obligatiune clasica
|  | Introduceti nr de cunostinte pentru acest obiect: 7
|  |
|  | Introduceti numele obiectului: obligatiune cu dobanda variabila
|  | Introduceti nr de cunostinte pentru acest obiect: 6
|  |
|  | Introduceti numele obiectului: obligatiune cu cupon zero
|  | Introduceti nr de cunostinte pentru acest obiect: 7
|  |
|  | Introduceti numele obiectului: obligatiune cu durata variabila
|  | Introduceti nr de cunostinte pentru acest obiect: 7
|  |
|  +-----+
|
|          ||
|          ||
|  +-----+
|
F2-Save  F3-Load  F6-Switch  F9-Compile          Alt-X-Exit

```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+-----Meniu baza de cunostinte-----+
|Introduceti numarul de caracteristici: 7
|
|Introduceti numele caracteristicii: durata
|Introduceti mesaj initial: Are durata
|Introduceti mesaj final: ?
|Aceasta caracteristica este exclusiva? (da/nu): da
|Cate valori avem pentru aceasta caracteristica? 3
|Introduceti numele valorii: nelimitata
|Introduceti numele valorii: limitata
|Introduceti numele valorii: variabila
|
|Introduceti numele caracteristicii: dividend
+|Introduceti mesaj initial: Are un dividend|+
+|Introduceti mesaj final: ?|+
|Aceasta caracteristica este exclusiva? (da/nu): nu
|+-----+-----+
|                                     ||
|                                     ||
+-----+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                               Alt-X-Exit

```

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+-----Meniu baza de cunostinte-----+
|Introduceti numele caracteristicii: risc
|Introduceti mesaj initial: Are un grad
|Introduceti mesaj final: de risc?
|Aceasta caracteristica este exclusiva? (da/nu): da
|Cate valori avem pentru aceasta caracteristica? 3
|Introduceti numele valorii: scazut
|Introduceti numele valorii: mediu
|Introduceti numele valorii: ridicat
|
|Introduceti numele caracteristicii: venit
|Introduceti mesaj initial: Asigura un venit
|Introduceti mesaj final: ?
|Aceasta caracteristica este exclusiva? (da/nu): nu
+|Cate valori avem pentru aceasta caracteristica? 4|+
+|Introduceti numele valorii: sigur|+
|Introduceti numele valorii: nesigur
|+-----+-----+
|                                     ||
|                                     ||
+-----+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                               Alt-X-Exit

```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+-----Meniu baza de cunostinte-----+
|Introduceti numele caracteristicii: dobanda
|Introduceti mesaj initial: Asigura o dobanda
|Introduceti mesaj final: ?
|Aceasta caracteristica este exclusiva? (da/nu): da
|Cate valori avem pentru aceasta caracteristica? 3
|Introduceti numele valorii: fixa
|Introduceti numele valorii: variabila
|Introduceti numele valorii: zero
|
|Introduceti numele caracteristicii: vot
|Introduceti mesaj initial: Fara drept de vot
|Introduceti mesaj final: ?
|Aceasta caracteristica este exclusiva? (da/nu): da
+|Cate valori avem pentru aceasta caracteristica? 2|+
+|Introduceti numele valorii: da|+
+|Introduceti numele valorii: nu|+
|+-----+
|
|
|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                      Alt-X-Exit

```

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+-----Meniu baza de cunostinte-----+
|*****
|
|Dati cunostintele despre entitatea obligatiune cu cupon zero
|*****
|Cod      caracterist
|*****
|1        contravaloare
|2        vot
|3        dobanda
|4        venit
|5        risc
|6        dividend
+|7        durata|+
+|*****|+
|Introduceti codul caracteristicii: 7
|+-----+
|
|
|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                      Alt-X-Exit

```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+-----Meniu baza de cunostinte-----+
|3      dobanda
|4      venit
|5      risc
|6      dividend
|7      durata
|*****
|Introduceti codul caracteristicii: 7
|
|*****
|Cod      Valoare
|*****
|1      variabila
|2      limitata
+|3      nelimitata
+|*****
+|Introduceti codul valorii: 2
+|
+|
+|
+|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                      Alt-X-Exit

```

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+-----Meniu baza de cunostinte-----+
|2      limitata
|3      nelimitata
|*****
|Introduceti codul valorii: 2
|*****
|Cod      caracterist
|*****
|1      contravaloare
|2      vot
|3      dobanda
|4      venit
|5      risc
|6      dividend
+|7      durata
+|*****
+|Introduceti codul caracteristicii: 5
+|
+|
+|
+|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                      Alt-X-Exit

```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
| +-----Meniu baza de cunostinte-----+
| |
| | 2      mediu
| | 3      scazut
| | *****
| | Introduceti codul valorii: 3
| | *****
| | Cod      caracterist
| | *****
| | 1      contravaloare
| | 2      vot
| | 3      dobanda
| | 4      venit
| | 5      risc
| | 6      dividend
| | 7      durata
| | *****
| | Introduceti codul caracteristicii: 4
| |
| |
| +-----+
|
|          ||
|          ||
| +-----+
F2-Save F3-Load F6-Switch F9-Compile Alt-X-Exit

```

```

Files          Edit          Run          Compile          Options          Setup
+-+-----+-----+-----+-----+-----+-----+
|                                     Meniu principal                                     |
|+-----+-----+-----+-----+-----+-----+|
|3      dobanda|
|4      venit  |
|5      risc   |
|6      dividend|
|7      durata |
|*****|
|Introduceti codul caracteristicii: 1|
|
|*****|
|Cod      Valoare|
|*****|
|1      bani    |
|2      bunuri  |
|*****|
|Introduceti codul valorii: 1|
|Am salvat baza de cunostinte.|
|+-----+-----+-----+-----+-----+-----+|
|                                     ||
|                                     ||
|+-----+-----+-----+-----+-----+-----+|
F2-Save  F3-Load  F6-Switch  F9-Compile
Alt-X-Exit

```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

[illegible][illegible]

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+----- Directory: .\*.CUN -----+
|  CAINI.CUN      LACURI.CUN      PLANTE.CUN      PRODFINC.CUN
|
|Am încarcat baza de cunostinte: .\PRODFINC.CUN.
|
|Are durata variabila?
|da/nu? nu
|
|Are durata nelimitata?
|da/nu? nu
|
+
+
|Are durata limitata?
|da/nu? da
+
+
|
|
|
|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                      Alt-X-Exit

```

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|+----- Directory: .\*.CUN -----+
|da/nu? da
|
|Are un grad scazut de risc?
|da/nu? da
|
|Asigura un venit sigur?
|da/nu? nu
|
|
|
|Asigura un venit variabil?
|da/nu? da
+
+
|Asigura o dobanda variabila?
|da/nu? da
+
+
|
|
|
|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                      Alt-X-Exit

```

Anexa 18 – Creare bază de cunoștințe și identificare produs financiar cu “Expert Generator”

```

Files      Edit      Run      Compile      Options      Setup
++-----Meniu principal-----++
|
|----- Directory: .\*.CUN -----|
|
|Asigura un venit variabil?
|da/nu? da
|
|Asigura o dobanda variabila?
|da/nu? da
|
|Fara drept de vot?
|da/nu? da
|
|Contravaloarea este in bani?
|da/nu? da
+
+
|Este produsul financiar obligatiune cu dobanda variabila!
+
+-----+
|
|
|
+-----+
F2-Save  F3-Load  F6-Switch  F9-Compile                                Alt-X-Exit

```

Anexa 19 – Exemplificare a bazei de cunoștințe în domeniul bursier creată cu “Expert Generator”

entitate("actiune preferentiala",8,6)
entitate("actiune obisnuita",7,7)
entitate("actiune de aport",6,7)
entitate("actiune cu dividend prioritar",5,7)
entitate("obligatiune clasica",4,7)
entitate("obligatiune cu dobanda variabila",3,6)
entitate("obligatiune cu cupon zero",2,7)
entitate("obligatiune cu durata variabila",1,7)
caracterist("durata",7,"Are durata","?")
caracterist("dividend",6,"Are un dividend","?")
caracterist("risc",5,"Are un grad","de risc?")
caracterist("venit",4,"Asigura un venit","?")
caracterist("dobanda",3,"Asigura o dobanda","?")
caracterist("vot",2,"Fara drept de vot","?")
caracterist("contravaloare",1,"Contravaloarea este in","?")
exclusiv(7)
exclusiv(5)
exclusiv(3)
exclusiv(2)
exclusiv(1)
valoare(7,"nelimitata",3)
valoare(7,"limitata",2)
valoare(7,"variabila",1)
valoare(6,"limitat",2)
valoare(6,"prioritar",1)
valoare(5,"scazut",3)
valoare(5,"mediu",2)
valoare(5,"ridicat",1)
valoare(4,"sigur",4)
valoare(4,"nesigur",3)
valoare(4,"variabil",2)
valoare(4,"fix",1)
valoare(3,"fixa",3)
valoare(3,"variabila",2)
valoare(3,"zero",1)
valoare(2,"nu",2)
valoare(2,"da",1)
valoare(1,"bunuri",2)
valoare(1,"bani",1)
legatura(1,7,1,1)
legatura(1,5,3,2)
legatura(1,4,4,3)
legatura(1,3,3,4)
legatura(1,2,1,5)
legatura(1,1,1,6)
legatura(1,4,1,7)
legatura(2,7,2,1)
legatura(2,5,3,2)
legatura(2,4,4,3)
legatura(2,4,1,4)
legatura(2,3,1,5)
legatura(2,2,1,6)
legatura(2,1,1,7)
legatura(3,7,2,1)
legatura(3,5,3,2)
legatura(3,4,2,3)

Anexa 19 – Exemplificare a bazei de cunoștințe în domeniul bursier creată cu “Expert Generator”

legatura(3,3,2,4)
legatura(3,2,1,5)
legatura(3,1,1,6)
legatura(4,7,2,1)
legatura(4,5,3,2)
legatura(4,4,4,3)
legatura(4,4,1,4)
legatura(4,3,3,5)
legatura(4,2,1,6)
legatura(4,1,1,7)
legatura(5,7,3,1)
legatura(5,5,2,2)
legatura(5,4,2,3)
legatura(5,4,3,4)
legatura(5,6,1,5)
legatura(5,2,1,6)
legatura(5,1,1,7)
legatura(6,7,3,1)
legatura(6,5,1,2)
legatura(6,4,2,3)
legatura(6,4,3,4)
legatura(6,6,2,5)
legatura(6,2,2,6)
legatura(6,1,1,7)
legatura(7,7,3,1)
legatura(7,5,1,2)
legatura(7,4,2,3)
legatura(7,4,3,4)
legatura(7,6,2,5)
legatura(7,2,2,6)
legatura(7,1,1,7)
legatura(8,7,1,1)
legatura(8,5,2,2)
legatura(8,4,2,3)
legatura(8,6,2,4)
legatura(8,2,1,5)
legatura(8,1,1,6)
comentariu_entitate("produsul financiar")

Anexa 20 – Sistem expert elaborat în Prolog

```
database
    is_fact(symbol,symbol)
    is_not(symbol,symbol)
predicates
    titlu
    titlu1
    init
    identifica(symbol)
    durata(symbol)
    dividend(symbol)
    vot(symbol)
    contravaloare(symbol)
    dobinda(symbol)
    venit(symbol)
    risc(symbol)
    out_message(symbol)
    run_again(symbol)
    act_on(symbol,symbol,symbol)
    redo
    forget
clauses
    titlu:-init,titlu1.
    titlu1:- identifica(Titlu),
               out_message(Titlu),
               forget.

init:-makewindow(1,7,108,"PRODUSE FINANCIARE",1,10,20,60),
      shiftwindow(1),
      write("Acest program permite identificarea"),nl,
      write("produselor financiare in vederea"),nl,
      write("folosirii disponibilitatilor banesti."),nl,
      write("Programul va pune o serie de intrebari"),nl,
      write("la care trebuie sa raspundeti cu 'da' sau 'nu'."),nl,
      write("Programul va incerca sa identifice"),nl,
      write("produsul financiar in care doriti "),nl,
      write("sa investiti."),nl,
      write("Apasati tasta ENTER pentru a continua:"),
      readchar(_),nl,nl.

identifica(actiune_preferentiala):- durata(variabila),
                                     dividend(limitat),
                                     vot(nu),
                                     contravaloare(bani),
                                     venit(variabil),
                                     risc(mediu).

identifica(actiune_obisnuita):- durata(nelimitata),
                                dividend(limitat),
                                vot(da),
                                contravaloare(bani),
                                venit(variabil),
                                venit(nesigur),
                                risc(ridicat).

identifica(actiune_de_aport):- durata(nelimitata),
                               dividend(limitat),
                               vot(da),
                               contravaloare(bunuri),
```

Anexa 20 – Sistem expert elaborat în Prolog

```
venit(nesigur),
venit(variabil),
risc(ridicat).

identifica(actiune_cu_dividend_prioritar):-durata(nelimitata),
    dividend(prioritar),
    vot(nu),
    contravaloare(bani),
    venit(nesigur),
    venit(variabil),
    risc(mediu).

identifica(obligatiune_clasica):-    durata(limitata),
    dobinda(fixa),
    vot(nu),
    contravaloare(bani),
    venit(sigur),
    venit(fix),
    risc(scazut).

identifica(obligatiune_cu_dobinda_variabila):-durata(limitata),
    dobinda(variabila),
    vot(nu),
    contravaloare(bani),
    venit(variabil),
    risc(scazut).

identifica(obligatiune_cu_cupon_zero):-durata(limitata),
    dobinda(zero),
    vot(nu),
    contravaloare(bani),
    venit(sigur),
    venit(fix),
    risc(scazut).

identifica(obligatiune_cu_durata_variabila):-durata(variabila),
    dobinda(fixa),
    vot(nu),
    contravaloare(bani),
    venit(sigur),
    venit(fix),
    risc(scazut).

identifica(no):-!.

durata(Tip):-is_fact(durata,Tip),!.
durata(Tip):-is_not(durata,Tip),!,fail.
durata(Tip):-clearwindow,nl,
    write("Doriti cu durata "),
    write(Tip),
    write("?"),
    readln(Response),
    act_on(Response,durata,Tip).

dividend(Tip):-is_fact(dividend,Tip),!.
dividend(Tip):-is_not(dividend,Tip),!,fail.
dividend(Tip):-clearwindow,nl,
    write("Doriti cu dividend "),
```

```

write(Tip),
write("?"),
readln(Response),
act_on(Response,dividend,Tip).

vot(Tip):-is_fact(vot,Tip),!.
vot(Tip):-is_not(vot,Tip),!,fail.
vot(Tip):-clearwindow,nl,
    write("Doriti sa "),
    write(" aveti drept de vot (" ,Tip,")?"),
    readln(Response),
    act_on(Response,vot,Tip).

contravaloare(Tip):-is_fact(contravaloare,Tip),!.
contravaloare(Tip):-is_not(contravaloare,Tip),!,fail.
contravaloare(Tip):-clearwindow,nl,
    write("Doriti sa platiti in "),
    write(Tip),
    write("?"),
    readln(Response),
    act_on(Response,contravaloare,Tip).

dobinda(Tip):-is_fact(dobinda,Tip),!.
dobinda(Tip):-is_not(dobinda,Tip),!,fail.
dobinda(Tip):-clearwindow,nl,
    write("Doriti cu dobinda "),
    write(Tip),
    write("?"),
    readln(Response),
    act_on(Response,dobinda,Tip).

venit(Tip):-is_fact(venit,Tip),!.
venit(Tip):-is_not(venit,Tip),!,fail.
venit(Tip):-clearwindow,nl,
    write("Doriti sa aveti un venit "),
    write(Tip),
    write("?"),
    readln(Response),
    act_on(Response,prima,Tip).

risc(Tip):-is_fact(risc,Tip),!.
risc(Tip):-is_not(risc,Tip),!,fail.
risc(Tip):-clearwindow,nl,
    write("Acceptati sa aveti un risc "),
    write(Tip),
    write("?"),
    readln(Response),
    act_on(Response,prima,Tip).

act_on(da,Attribute,Tip) :-
    asserta(is_fact(Attribute,Tip)).

act_on(nu,Attribute,Tip) :-
    asserta(is_not(Attribute,Tip)),fail.

out_message(no) :- !,nl,nl,
    write("*****"),
    nl,nl,

```

Anexa 20 – Sistem expert elaborat în Prolog

```
write("Eroare !"),nl,nl,
write("Nu poate fi identificat produsul financiar."),nl,nl.

out_message(Titlu) :- clearwindow,nl,nl,
    write("*****"),
    nl,nl,
    write("Produsul financiar este o "),
    write(Titlu),nl,nl.

forget :- retractall(is_fact(_,_)),
    retractall(is_not(_,_)),
    nl,write(" Program terminat. "),nl,nl,
    redo.

redo :- write("Folositi din nou programul ? "),
    readln(Response),
    run_again(Response).

run_again(da) :- nl,nl,!,
    titlu1.

run_again(nu) :- nl,nl.
```